#6

DERWENT ABSTRACT
KR 1999-0077851

10/9/1
DIALOG(R)File 351:Derwent WPI

012775963 **Image available**
WPI Acc No: 1999-582189/199950
XRPX Acc No: N99-430042
High speed remote storage cluster interface controller for computer system with several clusters of symmetric multiprocessors having interfaces between cluster nodes of symmetric multiprocessor system
Patent Assignee: INT BUSINESS MACHINES CORP (IBMC); IBM CORP (IBMC)
Inventor: BLAKE M A; MAK P; VANHUBEN G A; VAN HUBEN G A
Number of Countries: 030 Number of Patents: 006
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
EP 945798	A2	19990929	EP 99302070	A	19990317	199950 B
CA 2262314	A1	19990923	CA 2262314	A	19990222	200008
CN 1236136	A	19991124	CN 99104167	A	19990322	200014
US 6038651	A	20000314	US 9846430	A	19980323	200020
JP 2000076130	A	20000314	JP 9975531	A	19990319	200024
KR 99077851	A	19991025	KR 998429	A	19990312	200052

Priority Applications (No Type Date): US 9846430 A 19980323
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
EP 945798	A2	E	30	G06F-009/46	
	Designated States (Regional): AL AT BE CH CY DE DK ES FI FR GB GR IE IT LI LT LU LV MC MK NL PT RO SE SI				
CA 2262314	A1	E		G06F-015/173	
CN 1236136	A			G06F-015/16	
US 6038651	A			G06F-015/16	
JP 2000076130	A		34	G06F-012/06	
KR 99077851	A			G06F-015/16	

Abstract (Basic): EP 945798 A2

NOVELTY - The system has a remote resource manager working with a remote storage controller (10) to distribute work to a remote controller performing a desired operation without requiring knowledge of the initiator of the work request. Work is transferred only when a remote request is available to process it, without the need for constant communication between clusters of symmetric multiprocessors.

USE - For providing a high speed remote storage cluster interface controller for a computer system.

ADVANTAGE - Work is transferred only when a remote request is available to process it, without the need for constant communication between clusters of symmetric multiprocessors.

DESCRIPTION OF DRAWING(S) - The drawing shows a single storage controller cluster of a bi-nodal symmetric multiprocessor system.

the remote storage controller (10)

pp; 30 DwgNo 1A/8

Title Terms: HIGH; SPEED; REMOTE; STORAGE; CLUSTER; INTERFACE; CONTROL; COMPUTER; SYSTEM; CLUSTER; SYMMETRICAL; INTERFACE; CLUSTER; NODE; SYMMETRICAL; MULTIPROCESSOR; SYSTEM

Derwent Class: T01

International Patent Class (Main): G06F-009/46; G06F-012/06; G06F-015/16; G06F-015/173

International Patent Class (Additional): G06F-012/08; G06F-013/00

File Segment: EPI

Manual Codes (EPI/S-X): T01-F02; T01-F02C; T01-J05B3

?

#6

(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(51) Int. Cl.[6]
G06F 15/16

(11) 공개번호 특1999-0077851
(43) 공개일자 1999년10월25일

(21) 출원번호 10-1999-0008429
(22) 출원일자 1999년03월12일
(30) 우선권주장 9/046.430 1998년03월23일 미국(US)

(71) 출원인 인터내셔널 비지네스 머신즈 코포레이션 포만 제프리 엘
미국 10504 뉴욕주 아몬크

(72) 발명자 반후벤게리알렌
미국12603뉴욕주포우킵시에데일리로드22

블레이크마이클에이.
미국12590뉴욕주와핑거스폴스센트럴애비뉴3

막팍-킨
미국12603뉴욕주포우킵시에트로터레인7

(74) 대리인 장수길, 주성민

심사청구 : 없음

(54) 원격 자원 관리 시스템

요약

대칭적 멀티프로세싱에서 자원들을 관리하는 원격 자원 관리 시스템은 대칭적 멀티프로세서 시스템의 클러스터 노드들간에 인터페이스를 가진 대칭적 멀티프로세서의 다수의 클러스터 - 이들 클러스터의 각각은 로컬 인터페이스 및 인터페이스 제어기를 가짐 - 와, 제각기 로컬 인터페이스 제어기를 가진 하나 이상의 원격 저장 제어기와, 로컬-원격 데이터 버스와, 대칭적 멀티프로세서들의 두 클러스터간의 인터페이스를 관리하기 위한 원격 자원 관리기 - 상기 두 클러스터의 각각은 다수의 프로세서, 공유 캐쉬 메모리, 다수의 I/O 아답터 및 클러스터로부터 액세스 가능한 메인 메모리를 가짐 - 을 구비하며, 상기 원격 자원 관리기는 원격 저장 제어기를 가진 자원들을 관리하여 그 원격 제어기에 작업을 분배하며, 이 제어기는 작업 요구를 개시한 요구기(requester)를 알 필요 없이 원하는 동작을 수행하는 대리 프로세스(agent)로서 작용하고, 상기 작업은 대칭적 멀티프로세서들의 상기 클러스터들간의 일정한 통신을 필요로 하지 않고서도 원격 요구기가 그 작업을 처리하는데 이용가능할 때에만 전달된다.

대표도

도1a

색인어

멀티프로세싱, 자원 관리, 캐쉬 메모리, 로컬 인터페이스 제어기, 레지스터

명세서

도면의 간단한 설명

도 1a는 2-노드 대칭적인 멀티프로세서 시스템의 단일 저장 제어기 클러스터를 도시한 도면.

도 1b는 원격 클러스터로부터 복귀하는 모든 응답 통신 신호를 처리하는 응답 처리기와, 원격 저장 제어기 RSC IC(10)를 구성하는 세부적인 서브-유니트 및 인터페이스와, 원격 인출/저장 제어기(12)와 RSC IC간의 인터페이스를 도시한 도면.

도 2는 하드웨어로 명령 변환을 구현하는 양호한 실시예를 도시환 도면.

도 3a는 메인 명령 우선순위 스테이션(33)과 이것에 신호를 공급하는 파이프라인 사전 우선순위 스테이션(32)을 구비한 가진 전체적인 우선순위 스테이션을 도시한 도면.

도 3b는 상기한 메카니즘과 RSC IC 우선순위 스테이션(21)간의 상호작용을 보여주기 위한 로직 블록도.

도 4는 적당한 자원 보유 레지스터내로의 명령 스테이징 방법을 도시한 도면.

도 5는 디스에이블 비트, 유효 비트, 명령 레지스터, 오리지널 요구기 ID 레자스터, LFSAR 제어기 ID 레지스터 및 LFSAR 버퍼 레지스터를 제공하는 단일 세트의 자원 레지스터(59a - 59f)를 상세히 도시한 도면.

도 6은 스테이징 메카니즘과 응답 처리기(22)가 그 스테이징 메카니즘을 사용해서 XI 응답을 추적하는 방법을 도시한 도면.

도 7은 달리 설명되는 스테이징 파이프라인과 유사한 방식으로 작동하는 판독 전용 무효화 스테이징 파이프라인(75)을 도시한 도면.

도 8은 한 파이프라인이 한 RSC IC내에서 크로스포인트를 수신하고 구동하는 내부 로직을 도시한 도면.

<도면의 주요부분에 대한 부호의 설명>

10 : RSC IC
11 : 로컬 인출/저장 제어기
12 : 원격 인출/저장 제어기
12a : 원격 인출 제어기(RFAR)
12b : 원격 저장 제어기(RSAR)
13 : 파이프라인
15 : CPU 제어기
21 : RSC 우선순위 스테이션
22 : 응답 처리기
23 : 응답 우선순위 스테이션
24 : RSC 자원 레지스터
25 : 크로스포인트(XPT) 제어기
26 : 플립 비트 발생기
27 : XLAT
28 : 명령 변환 테이블
32 : 사전 우선순위 스테이션
33 : 메인 명령 우선순위 스테이션
35 : 자원 토글러
36 : 토글 보유 래치
37 : 토글러 진리표
41 : 3-웨이 멀티플렉서
42 : CLC 명령 스테이징 레지스터
43 : C3 파이프 명령 스테이징 레지스터
44 : 2-웨이 멀티플렉서
45 : RFAR 명령 레지스터
47 : RSAR 명령 레지스터
46 : 게이트웨이
52a : 스테이징된 신속-경로 승인 LC
53 : CLC 승인
54 : OR 게이트
55a : C3 신속-경로 스테이징 레지스터
55b : CLC 스테이징 레지스터
56 : 2-웨이 멀티플렉서
57 : 인코더
58 : 게이트웨이
59 : 자원 레지스터
59a : 자원 유효 비트
59b : 명령
59c : 요구기 ID
59d : LFSAR 버퍼
59e : LFSAR ID
59f : 디스에이블 비트
61 : 디코더
63 : AND/OR 멀티플렉서
64 : 4-비트 L3 스테이징 파이프라인
65 : 자원 로드 래치

66 : 8/3 인코더
67 : RFSAR 스테이징 파이프라인
68 : LFSAR 룩업 테이블
73 : ROI 디코더
74 : ROI 멀티플렉서
75 : 판독 전용 무효화 스테이징 파이프라인
81a, 81b : XPT GEN 로직
82a, 82b : 트리거 로직
83 : XPT_CNTR(83)

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 컴퓨터 시스템에 관한 것으로서, 특히 고속 원격 저장 클러스터 인터페이스 제어기(high speed remote storage cluster interface controller)에 관한 것이다.

역사적으로, 시스템 설계자들은 다양한 방법을 이용해서 다수의 긴밀하게 결합된 대칭적 멀티프로세서(tightly coupled symmetrical multiprocessor: SMP)에서 고성능을 구현해 왔는데, 이들은 단일의 공유 시스템 버스를 통해 상호결합하는 개별 프로세서들 및 프로세서 클러스터들(individual processors or processor clusters)에서부터 한 클러스터내에 있는 상호결합 프로세서(coupling processors)들에 이르기까지 다양하므로, 클러스터들은 클러스터-클러스터 인터페이스를 사용하여 중앙집중식 상호연결 네트워크(centrally interconnected network)와 통신을 행한다. 중앙집중식 상호연결 네트워크에서는, 다수(즉, 32 내지 1024)의 프로세서 주변에 구축된 병렬 시스템들이 중앙 스위치(즉, 크로스바 스위치(cross-bar switch)를 통해 상호연결된다.

공유 버스 방식에 의하면, 단일의 버스 프로토콜(protocol)로 여러 유형의 자원(resource)들에 대한 서비스를 행할 수 있기 때문에 가장 효과적인 비용으로 시스템을 설계할 수 있고, 또한 버스에 부가적인 프로세서, 클러스터 또는 주변 장치를 부착시켜 시스템을 경제적으로 확장시킬 수 있다. 그러나, 많은 시스템에서, 중재 오버헤드(arbitration overhead)와 결합된 시스템 버스상의 통신 폭주 현상(congestion)으로 인해서 전반적인 시스템 성능이 저하되고 SMP 효율이 낮아지는 경향이 있다. 이들 문제점은 500㎒ 이상의 주파수로 동작하는 프로세서들을 이용하는 대칭적 멀티프로세서 시스템의 경우에 상당한 위험요인으로 될 수 있다.

중앙집중식 상호연결 시스템에 의하면, 그 시스템의 모든 프로세서에 공유된 자원에 대한 대기 시간이 동일하게 되는 장점이 제공된다. 이상적인 시스템에서는, 대기 시간이 동일하면 그 시스템 구조 또는 메모리 계층(hierarchy)에 대한 사전 지식 없이도 다수의 응용 또는 한 응용내의 병렬 스레드(parallel threads)를 이용가능한 프로세서들간에 분배시킬 수 있다. 이들 유형의 시스템은 일반적으로 하나 이상의 대형 크로스바 스위치를 사용하여 프로세서들과 메모리간의 데이터 라우팅을 행함으로써 구현된다. 그러나, 이러한 시스템의 기반 구성에서는, 핀 패키징 요건(pin packing requirement)이 증대되고 고가의 부품 패키징이 필요하게 되며 또한 효율적인 공유 캐쉬(cache) 구조를 구현하는 곤란하게 된다.

긴밀하게 결합된 클러스터링 방식은 절충적인 해결책으로서, 여기서, 클러스터란 용어는 단일의 메인 메모리를 공유하는 프로세서들의 집단을 의미하므로, 특정 클러스터에 대한 친밀성에 관계없이 그 시스템내의 어떠한 프로세서도 메인 메모리의 어떠한 부분도 액세스 할 수 있다. 비균일 메모리 액세스(Non-Uniform Memory Access: NUMA)와는 달리, 여기서의 클러스터들은 전용 하드웨어를 사용하여 각 클러스터내에 있는 메모리와 제 2 레벨 캐쉬간의 데이터 코히어런시(coherency)를 유지시킴으로써, 어떠한 메모리 계층이나 물리적 분할 예를 들어 메모리 뱅크 인터리브(memory bank interleave)도 없이 통합된 단일의 이미지를 소프트웨어에 제공한다. 이들 시스템의 한가지 장점을 말하자면, 데이터를 필요로 하는 프로세서들에 근접하게 데이터가 유지되는 경우 예를 들어 데이터가 클러스터의 제 2 레벨 캐쉬나 그 클러스터에 부착된 메모리 뱅크 인터리브내에 있는 경우 클러스터내의 긴밀 결합된 프로세서들이 우수한 성능을 제공한다는 것이다. 또한, 이들 시스템은 중앙집중식 상호연결 시스템에 볼 수 있는 대형 N-웨이 크로스바 스위치에 비해 더욱 효과적인 비용의 패키징을 가능케 한다. 그러나, 이러한 클러스터링 방식에서는, 프로세서들이 다른 클러스터들로부터의 데이터를 빈번하게 요구하는 경우에 성능이 불량하게 되며, 또한 그 결과로서 생기는 대기 시간이 상당히 크게 되거나 대역폭이 부적절하게 된다.

중앙집중식 상호연결 시스템에 관련된 고비용의 문제점들의 대다수를 비용적인 면에서 효과적인 방식으로 해결할 수 있을 때까지는, 시장에서는 공유 버스 주변에 구축한 경제적인 시스템 또는 클러스터 구성이 지속적으로 요구될 것이다. 본 발명은 전통적인 클러스터 인터페이스 구성에 관련된 대다수의 결점을 제거하여 그 시스템이 프로세서 성능을 극대화하면서도 고비용의 하이 레벨 패키지 또는 고비용의 온-보드 캐쉬(on-board cache)를 필요 없게 한다. 본 발명과 관련된 이 분야의 종래기술에서는, 비용적인 면에서 효과적인 고주파수 저장 제어기의 설계에 관련된 전반적인 문제점 중의 특정 문제점을 해결하는 다양한 방안을 제시하고 있으나, 이들은 다음의 예에서 볼 수 있는 바와 같이 본 발명의 목적들을 충족하는 완전한 해결 방안을 제공하지 못한다.

대칭적 멀티프로세서의 두 클러스터로 구성된 시스템이 (1985년 3월 5일자로 특허된 크라이고스키(Krygowski)등의) 미국 특허 제4503497호에 개시되고 있다. 이특허에서는, 전용 스토어-인 캐쉬(private store-in cache)를 가진 프로세서들간의 캐쉬 코히어런시를 유지시키는 개량된 방법을 개시하고 있다. 그러나, 이 특허는 클러스터내에 있으나 그 클러스터에 연결된 모든 프로세서들이 공유하는 스토어-인 파이프라인드 레벨 2(store-in pipelined Level 2(L2)) 캐쉬와 연관된 다양한 문제를 다루고 있지 않다. 이 특

허는 또한 모든 유형의 동작(프로세서, I/O, 메모리, 방송 신호전송(broadcast signaling), 크로스 클러스터 동기화(cross cluster synchronization) 등등)을 위해 클러스터 인터페이스의 전체 효율을 극대화하는 것에 대해서 다루고 있지 않다.

초대형 SMP 시스템의 일 예는 1992년 12월 1일자로 특허된 밀러(Miller) 등의 미국 특허 제5,168,547호 및 1993년 3월 23일자로 특허된 첸(Chen) 등의 미국 특허 제5,197,130호에 개시되고 있다. 이들 특허는 제각기 다수(즉 32)의 프로세서 및 외부 인터페이스 수단을 가진 다수의 클러스터로 구성된 컴퓨터 시스템을 개시한다. 여기서의 각 프로세서는 모든 클러스터들에서 모든 공유된 자원들을 대칭적으로 액세스한다. 상기한 특허들의 컴퓨터 시스템은 그의 성능상의 목적을 대형 크로스바 스위치들의 조합, 고인터리브형(highly interleaved) 공유 메모리, 소스(source)와 목적지(destination)간의 경로가 이용가능하게 될 때까지의 스테이지 트랜잭션(stage transactions)에 대한 일련의 인바운드 및 아웃바운드 큐(inbound and outbound queues) 및 동기화 및 데이터 공유에 사용되는 클러스터 중재 수단내의 글로벌 자원 세트(a set of global resources)를 사용해서 달성한다. 이들 특허는 또한 다수의 병렬 프로세서간에 작업(work)을 보다 효율적으로 분할하는 방법을 구현하기 위해 (제 2 레벨 캐쉬들을 포함하는) 계층적 메모리 시스템을 사용할 필요가 없는 구성을 개시한다.

다수의 I/O 장치를 클러스터링하고 그들 장치를 인공 지능적 제어기(intelligent controller)로 관리하는 것에 의해서 전체 시스템 성능을 향상시키기 위한 수개의 방법이 또한 제시되고 있다. 1979년 5월 29자로 특허된 로링즈(Rawlings) 등의 미국 특허 제4,156,907호 및 1980년 4월 29일자로 특허된 로링즈 등의 미국 특허 제4,200,930호는 호스트 시스템으로부터의 데이터 및 메시지 전달을 오프로딩(offloading)하는 펌웨어 인에블드 I/O 프로세서(firm enabled I/O processors)를 구비한 개량된 아답터 클러스터 모듈 및 데이터 통신 서브시스템(Adapter cluster module and data communication subsystem)을 개시하고 있다. 이들 특허의 발명은 무수히 많은 전송 프로토콜을 사용하여 다양한 원격 주변 장치와 인터페이싱할 수 있다. 아답터 클러스터 모듈은 상호 공통점이 없는 프로토콜 하에서 동작하는 "바이트" 트래픽("byte" traffic)을 호스트 시스템에 단일 프로토콜에 의해서 효율적으로 전송될 수 있는 전체 메시지로 변환하는 것에 주로 관련된다. 이들 발명에서는 또한 통신 서브시스템이 호스트 시스템의 정지 시에도 원격 주변 전송을 지속적으로 처리할 수 있게 하는 여러 신뢰성 및 가용성을 활용한다. 여기서 개시하는 기법들은 I/O 서브시스템 레벨에서의 성능 문제를 분명 향상시키지만, 호스트 컴퓨터 시스템에서 메인 메모리와 두 프로세서 또는 한 프로세서간의 고속 데이터 전달이 필요함에 대해 다루고 있지 않다.

본 발명에 의해서 해결되는 전체적인 문제점 중의 어떤 것들을 다루고 있는 발명으로서는 몇 개가 있으나, 이들 발명 중의 그 어느 것도 그들 문제점 모두를 다루고 있지 않다. 특히 중요한 것은 이들 발명에서 개시하고 있는 사상들을 조합해도 본 발명에 의해서 제공되는 전체적인 효율성이 나타나지 않는다. 예를 들어, (1995년 2월 21일자로 특허된 바루시(Barucchi) 등의) 미국 특허 제5392401호에는 두 프로세서간에서 데이터를 전달하는 향상된 방법이 개시되고 있으나, 이 특허의 발명은 크로스바 스위치를 사용하는 것으로서 공유된 제 2 레벨 캐쉬의 캐쉬 코히어런스를 개시하고 있지 않다. (1984년 4월 24일자로 특허된 프레쳐(Fretcher)의) 미국 특허 제4445174호는 전용 캐쉬를 가진 프로세서들과 공유 레벨 2(L2)를 인터록킹하는 수단을 개시하고 있으나, 클러스터-클러스터 인터페이스와 연관된 대역폭 및 대기 시간의 문제를 다루고 있지 않다. (1993년 2월 9일자로 특허된 치나스워미(Chinnaswamy) 등의) 미국 특허 제5185875호는 데이터를 캐쉬내로 로딩함과 병행하여 요구된 프로세서(the requested processor)에 데이터를 라우팅하는 것에 의해 저장 제어 유니트들간의 데이터 전달 대기 시간을 감소시키는 방법을 개시하고 있다. 이와 유사한 기법들이 오늘날의 컴퓨터 시스템 구성에 널리 사용되고 있지만, 이 특허의 발명은 저장 제어 유니트가 캐쉬에 대한 액세스를 요구하는 (I/O 및 메모리를 포함하는) 각 시스템 자원용의 전용 핀 인터페이스를 제공할 수 없는 경우에 발생되는 문제점들을 해결하고 있지 못하다. (1988년 11월 15일자로 특허된 키리(Keely)의) 미국 특허 제4785395호는 적어도 한 쌍의 프로세서들간에 캐쉬를 공유하기 위한 방법을 개시하고 있으나, 이 방법은 모든 프로세서들이 동일한 대기 시간으로 캐쉬를 액세스할 수 있음을 가정한 것이다.

수개의 발명들이 개별 프로세서들 또는 프로세서들의 클러스터가 메인 메모리 및 외부 I/O 디바이스와 공유 버스를 통해 통신하는 경우 공유 버스 시스템내의 통신량을 중재하기 위한 기법들을 개시하고 있다. 예로서, (1988년 11월 15일자로 특허된 피셔(Fischer)의) 미국 특허 제4,785,394호는 공유 버스의 사용을 중재하는 방법을 개시하고 있는데, 이들의 기법은 응답기 우선권을 개시 프로그램(initiator)에 인계하여 수신 모듈이 사용중인 경우에도 그 수신 모듈에 대해 요구들이 개시될 수 있게 한다. 이 발명은 이러한 중재 동작을 원격지에 있는 자원들이 작업을 수용할 수 있는 경우에만 클러스터-클러스터를 작동하게 하는 것에 의해 향상시킨다. 또한, 응답기와 개시 프로그램간의 중재를 우선권을 고정하지 않는 상태로 매 사이클마다 동적으로 수행한다. (1986년 2월 11일자로 특허된 테트릭(Tetrick)의) 미국 특허 제4570220호는 시스템 버스의 절충을 위해 직렬 버스와 병렬 버스의 조합을 이용한다. 이 시스템 버스는 수개의 "대리 프로세스(agent)"간에 공유되는데, 이 경우 대리 프로세스는 핸드쉐이킹 시퀀스(handshaking sequence)와 정합하여 버스를 사용할 권리를 획득해야만 한다. 이 발명은 원격 자원들을 추적함으로써 어떤 유형의 버스 협상도 수행할 필요 없이 단일 클럭 사이클상에서 새로운 요구들을 동적으로 개시시킬 수 있다.

발명이 이루고자하는 기술적 과제

본 발명은 2-노드 SMP 시스템에서 두 클러스터간의 인터페이스를 관리하는 수단을 개시한다. 본 발명의 양호한 실시예는 제각기 전용 L1 캐쉬, 다수의 I/O 아답터(Adapter) 및 메인(main) 메모리를 가진 다수의 중앙처리기(Central Processor : CP)를 구비하는 대칭적 멀티프로세싱 시스템(Symmetric Multiprocessing System)내에 구현된다. 여기서는, 어떠한 프로세서나 I/O 아답터도 메인 메모리의 어떤 부분도 액세스할 수 있다. 전체 수의 프로세서와 I/O 아답터는 두 개의 클러스터로 동등하게 나누어 진다. 또한, 메인 메모리는 뱅크 또는 인터리브(banks or interleaves)로 이루어지며, 이들의 절반은 각 클러스터에 부착된다.

각 클러스터에는 저장 제어기(Storage Controller)가 존재하는데, 이 제어기는 공유된 제 2 레벨 캐쉬, 다양한 제어기 및 각 프로세서에 대한 개별 인터페이스(또는 포트), I/O 아답터 및 메인 메모리로 구성된다. 본 실시예의 캐쉬는 다수의 뱅크 또는 인터리브로 구성되며 그의 내용은 8-웨이 어소시에티브 디렉토리(8-way

associative directory)에 의해서 관리된다. 도 1a에 도시한 저장 제어기는 주요 기능 요소들을 나타내고 있는데, 이에 대해서는 양호한 실시예의 상세한 설명에서 설명하겠다. 그러나, 본 발명의 특징들을 이해하는데 있어서 도움을 위해 단일 클러스터내의 저장 제어기에 대해 간단히 개관해 보고자 한다.

저장 제어기의 주 기능은 메인 메모리에 관한 프로세서 및 I/O 아답터로부터의 데이터 인출 및 저장 요구들을 처리하고자 하는 것이다. 저장 제어기는 공유된 제 2 레벨 캐쉬를 포함하는데, 이 캐쉬는 소프트웨어 및 운영 시스템에 대해서는 구조적으로 눈에 보이지 않는다. 저장 제어기는 디렉토리 및 캐쉬 액세스를 수행하는 임무를 담당한다. 모든 입력 요구는 저장 제어기의 포트로 입력되고, 이 제어기에서 그들 요구는 중앙 처리기(CFAR) 또는 I/O 제어기에 의해 수신된다. 이들 제어기는 중앙 우선순위 유니트(Central Priority Unit)로 요구를 발생하고, 중앙 우선순위 유니트는 그들 요구를 중재하고 요구하는 요구기(requester)중의 하나를 선택하여 어드레스에 기초해서 두 멀티스테이지 파이프라인(multistage Pipeline)중의 하나에 입력시킨다. 파이프라인의 각 스테이지 동안, 요구기는 각종 자원 예를 들어 캐쉬, 로컬 캐쉬 인출/저장 제어기(Local Cache Fetch/Store Controller), 데이터 경로 제어기, 데이터 경로 선입선출 버퍼, 원격 캐쉬 인출/저장 제어기(Remote Cache Fetch/Store Controller) 등을 액세스하고/하거나 확보해 둔다.

요구가 파이프라인을 나올 때, 로컬 인출/저장 제어기(Local Fetch/Store Controller)중의 하나는 목적 달성 시까지 동작을 관리하는 임무를 담당한다. 이렇게 하는데에는 간혹 추가적인 파이프라인의 통과가 필요하므로, 로컬 인출/저장 제어기도 중앙 우선순위 중재에 참여해야 하며 요구기도 고려된다. 본 실시예에서는, 캐쉬 제어기 및 메인 메모리 제어기가 로컬 인출/저장 제어기의 일부로서 포함된다. 이들간에는 캐쉬 인터리브로부터 데이터를 액세스하고, 캐쉬의 적중실패(miss) 발생 시에 메인 메모리에 대한 데이터 액세스를 처리하고, 캐쉬 인터리브내로의 저장 동작을 수행하며, 메인 메모리 액세스로부터의 입력 데이터를 위한 룸(room)을 만들기 위해 캐쉬로부터 메인 메모리내로 오래된 데이터를 (최소 최근 사용 방법(Least Recently Used method)을 사용하여) 방출하는데 필요한 (데이터 경로 요소 예를 들어 선입선출 버퍼 및 크로스포인트 스위치를 포함하는) 모든 자원이 포함된다.

상술한 바와 같이, 메인 메모리 뱅크들은 2-노드 시스템의 두 클러스터간에 물리적으로 분포된다. 그러나, 메인 메모리는 SMP 시스템내의 어느 곳에 위치하는 프로세서들 또는 I/O 아답터들 중의 어떤 것에 대해서도 단일의 통합된 엔티티처럼 보인다. 그러므로, 본 실시예는 원격 인출/저장 제어기(Remote Fetch/Store Controller)로서 알려진 부가적인 세트의 제어기를 구비한다. 저장 제어기는 각 클러스터상의 메모리 뱅크에 어떤 메인 메모리 어드레스가 할당되는 가를 지속적으로 추적한다. 데이터 액세스(인출 요구)의 로컬 클러스터상의 캐쉬에 대한 적중실패가 발생될 때마다, (여기서 로컬이란 용어는 요구 발생 프로세서 또는 I/O 아답터가 부착된 클러스터를 말한다.), 로컬 인출/저장 제어기는 원격(또는"다른") 클러스터에 대한 질의를 행하여 그 캐쉬내에 데이터가 있는지를 알아 보아야 한다. 이들 원격 질의는 원격 인출 제어기에 의해 처리되며, 이 원격 인출 제어기는 중앙 우선순위 유니트내로 요구를 발생하고 로컬 인출/저장 제어기에 대해 유사한 방식으로 자원을 액세스한다.

또한, 데이터 액세스의 원격 캐쉬에 대한 적중실패가 발생되나, 어드레스가 원격 클러스터에 부착된 메모리 뱅크에 속함을 나타내면, 원격 인출/저장 제어기는 또한 메인 메모리 제어기와 상호작용을 행하여 메인 메모리 액세스를 개시한다. (오래된 데이터를 캐쉬 밖으로 방출하는 것과 같은) 메모리내로의 데이터 저장을 필요로 하는 동작의 경우, 어드레스는 로컬 인출/저장 제어기가 전체 동작을 처리할 수 있는지의 여부 또는 원격 저장 동작이 2-노드 인터페이스를 통해 개시되어야 하는지의 여부를 다시 한번 판단한다. 이러한 상황에서, 원격 저장 동작은 메모리 인터리브내로의 데이터 저장을 위해 메인 메모리 제어기와의 상호작용을 또한 행하는 원격 저장 제어기에 의해서 처리된다. 로컬 인출/저장 제어기의 경우와 같이, 그들의 원격 카운터파트(remote counterpart)는 클러스터간 동작을 처리하는데 필요한 (데이터 경로, 선입선출 버퍼 및 크로스포인트 스위치를 포함하는) 모든 자원을 또한 포함한다.

본 발명은 상기한 원격 인출/저장 제어기를 포함하는 자원들을 관리하여 그들 원격 인출/저장 제어기에 작업을 분배하기 위한 원격 관리 시스템에 관한 것이다. 이 제어기는 작업 요구를 개시한 요구기를 알 필요 없이 원하는 동작을 수행하는 대리 프로세스로서 작용한다. 작업은 대칭적 멀티프로세서의 다수 클러스터간의 일정한 통신을 필요로 하지 않고서도 원격 자원들이 그 작업을 처리하는데 이용가능할 때에만 분배된다. 이 시스템에서는 최소한의 인터페이스 통신 신호가 이용된다.

본 발명의 원격 자원 관리 시스템은 적은 수의 입력 및 출력 핀을 사용하여 높은 효율로 대칭적 멀티프로세서의 두 클러스터간의 인터페이스를 관리한다. 또한, 수개의 기법을 이용하여 핀 제약 요인을 극복하며 S/390 엔터프라이즈 서버(Enterprise Server)와 같은 매우 복잡한 컴퓨터 시스템내의 집적을 가능케 한다. 이 서버에서는 단일의 클러스터가 다수의 아주 높은 주파수의 프로세서, 공유 레벨 2 캐쉬, 수개의 I/O 아답터 수단 및 메인 메모리를 포함할 수 있다. 이러한 시스템에서는, 성능이 탁월하며, 캐쉬 적중실패와 관련된 대기 시간이 최소화된다. 따라서, 본 발명은 전체적인 시스템 성능을 최대화하면서 패키징 비용을 최소화시키고자 하는 것이다.

우선, 각 클러스터상의 단일 인터페이스 유니트는 인터페이스의 완전한 제어를 담당하는데, 여기에는 대기 요구들(queued requests)을 우선순위화하고, 새로운 동작들을 인터페이스를 통해 전송하고, 다른 사이드로부터의 복귀 응답을 처리하며, 클러스터들간에서의 모든 데이터 전송을 감독하는 것이 포함된다. 제어 I/O의 수가 제한되기 때문에, 본 발명에서는 명령 재맵핑과 원격 자원 관리의 새로운 조합을 사용해서 전송할 필요가 있는 정보의 양을 최소화한다. 로컬 인터페이스 제어기는 원격 사이드에 대해 작업 요구를 개시할 뿐만 아니라 원격 사이드상의 인출/저장 제어기를 관리하여, 이용가능한 제어기에 새로운 동작을 즉시 라우팅한다. 원격 인출/저장 제어기는 단순히 로컬 인터페이스 제어기 대신에 작업을 행하는 대리 프로세스로서 된다. 로컬 인터페이스 제어기는 요구기 대신에 작업을 행한다. 이러한 식으로 동작함으로써, 동작의 오너(owner)를 식별하는 정보를 보낼 필요가 없게 되는데, 이는 원격 사이드가 알아야 할 이유가 없기 때문이다.

원격 제어기는 수개의 로컬 동작이 단일의 단위(atomic) 원격 동작으로 조합될 수 있게 하는 명령 재맵핑 동작을 통해 더욱 간단하게 된다. 예를 들어, 판독 전용 데이터 카피(read-only copy of data)를 위한 프로세서 인출 요구 및 저장 보호 키(storage protection key)를 포함하는 판독 전용 데이터를 위한 인출 요구는 동일한 상태도 및 캐쉬 관리 동작을 이용하기 위해 원격 클러스터상의 인출 제어기를 필요로 한다. 그러므로, 인터페

이스 제어기는 그들 양자를 판독 전용 라인 인출(Read Only Line Fetch)로서 알려진 간단한 원격 저장 클러스터(RSC) 인터페이스 제어기 명령으로 재맵핑시켜, 원격 저장 클러스터(RSC) 인터페이스 제어기에 의해 처리되어야 하는 동작들의 수를 감소시킬 것이다.

이 재맵핑 동작의 부가적인 장점은 불필요한 데이터 전송을 배제시키는 것에 의해 인터페이스 경로들을 더욱 효율적으로 관리할 수 있다는 것이다. 예인 메모리로의 저장되기 전에 동일 라인 데이터의 가장 최근 카피와 합병될 입력 64개 바이트들을 필요로 하는 64-바이트 I/O 저장에 대해 고려한다. 이 동작에 의하면, 목표 메인 저장 어드레스(target main store memory)와 현재 캐쉬 상태에 따른 다음과 같은 세가지의 서로 다른 시나리오가 생길 수 있다.

1. 데이터가 원격 사이드상의 메인 메모리를 목표로 하고 있고 로컬 캐쉬의 적중실패가 발생한 경우에는, I/O 저장 데이터는 합병을 위해 다른 사이드로 보내져야 하는데, 이를 위해서는 로컬 클러스터로부터 원격 클러스터로의 저장 동작을 수행하는 RSC 인터페이스 제어기(RSC IC)가 필요하게 된다.

2. 데이터가 로컬 메모리를 목표하고 있으나 원격 캐쉬에서 적중(hit)한 경우에는, 원격 사이드로부터 라인을 검색하여 로컬 클러스터상에서 합병이 발생하도록 할 필요가 있다. 이렇게 하기 위해서는, 가능한 데이터 인출과 함께 원격 사이드에 대한 교차 질의가 필요하다.

3. 라인의 카피가 그들 두 캐쉬에 존재하는 경우에는, 단지 요구되는 조치는 원격 사이드내의 라인을 무효화하는 것인데, 이는 입력 64-바이트가 로컬 캐쉬내의 카피와 합병될 수 있기 때문이다.

더욱 간단한 구성은 I/O 저장 명령을 64-바이트 데이터와 함께 조건없이 인터페이스를 통해 전송하고, 다른 사이드의 원격 인출/저장 제어기가 그때 디렉토리 상태에 근거해서 필요한 조치를 수행하는 것일 것이다. 그러나, 상기한 세가지 경우 중의 두가지 경우에서, 저장 데이터의 전송은 로컬-원격 데이터 버스를 불필요하게 구속할 것이다. 또한, 디렉토리 정보의 전송을 위해서 부가적인 제어 라인들이 필요하게 될 것이다. 본 발명에서는 상기한 세가지 경우 중의 나중의 두가지 경우를 제각기 "강제 방출(force cast out)" 및 "판독 전용 무효화(read-only invalidate)" 명령으로 재맵핑하는 인공 지능적 인터페이스 제어기를 이용한다.

명령 재맵핑에 의하면, 수개의 장점이 제공된다. 첫째, 많은 동작이 더욱 간단한 단위 인터페이스 동작으로 맵핑되므로 원격 인출/저장 제어기 구성이 간단해 진다. 둘째, 클러스터간의 디렉토리 정보 전송을 위해 어떠한 부가적인 제어 I/O도 필요하지 않게 된다. 셋째, 어떠한 대기 시간의 증가도 방지되도록, 새로운 명령이 인터페이스를 횡단하게 하기 위한 우선순위가 동일한 사이클내에서 발생되는 명령 재맵핑이 수행된다.

원격 관리 시스템은 단일 또는 다수의 파이프라인드 레벨 2 캐쉬에 대해 서비스하는 다수의 인출 및 저장 제어기를 포함하는 하이-엔드(high-end) 저장 서브시스템과 인터페이스되도록 구성된다. 일련의 우선순위 스테이션은 인터페이스를 통해 전송하기 위한 요구를 궁극적으로 선택하는데 사용된다. 다수의 파이프가 포함되는 경우, 각 파이프내의 사전 우선순위 스테이션은 RSC IC로 전송하기 위한 인출 또는 저장 요구중의 하나를 선택한다. 동일 사이클 동안, RSC IC는 명령 유형 및 자원 가용성에 근거하여 최적의 요구를 선택하기 위해 고성능 우선순위 동작을 이용한다. 다수의 파이프가 어떤 주어진 사이클에서 인터페이스 사용을 요구할 수 있으므로, 원격 인출 제어기를 이용할 수 있는 동안에는 동작은 저장보다 인출을 지지할 것이다. 그렇지 않으면, 원격 저장 제어기가 이용가능하고 또한 데이터 경로가 그를 필요로 하는 저장 동작에 대해 이용가능한 경우에는 저장이 취해질 것이다. 그들 두 요구가 인출이고 그들 두 요구가 이용가능한 자원을 갖는 경우에는, 어떤 요구를 우대할 것인지의 여부를 간단한 라운드 로빈(round robin) 방법에 의해 판단한다. 그들 두 요구가 저장인 경우에는, 어느 파이프가 이용가능한 자원을 갖는 가에 의해 승자가 결정된다. 또한, 그들 둘 모두가 이용가능한 자원을 갖는 경우에는 간단한 라운드 로빈 방법을 사용한다. 이 방법에 의하면, 작업 요구 및 이용가능 자원이 있는 동안에는 사실상 명령이 전송될 것이다. 또한, 프로세서에 제공되는 우선적인 처리에 의해서 전체적인 시스템 성능이 향상된다. 마지막으로, 로컬 인터페이스 제어기내의 원격 자원을 관리함으로써, 원격 사이드상에 대기되고 있는 작업을 전송하는 인터페이스 사이클이 허비되지 않을 것이다.

L1 캐쉬 적중실패로 인한 프로세서 데이터 액세스 대기 시간을 더욱 감소시키기 위해, RSC IC는 어떠한 인출 또는 저장 제어기도 인터페이스의 사용을 요구하고 있지 않는 사이클 동안 "신속-경로화(fast-pathing)" 기법을 이용한다. 이들 사이클 동안, 모든 파이프는 유효 CP 인출에 관해 감시된다. 이 인출이 발견되면, 그것이 원격 사이드로 즉시 디스패칭(dispatching)되며, 한편 이와 병행해서 로컬 캐쉬 인출 제어기가 로딩된다. 따라서, 인출 요구가 원격 사이드로 한 사이클 앞서 출발할 수 있어 복귀 데이터의 대기 시간이 감소될 수 있다.

원격 캐쉬에 대해 적중하는 데이터 인출과 로컬 메인 메모리로부터의 데이터 액세스는 최상 경우의 대기시간에 상당한 차이가 있기 때문에, RSC IC는 원격 캐쉬 적중을 로컬 인출 제어기로 알릴 수 있어, 메인 저장 액세스가 취소될 수 있다. 따라서, 다른 요구에 대한 작업을 위해 메모리 뱅크들을 자유롭게 할 수 있어 전체 시스템 성능이 향상된다.

RSC IC는 동기식 및 비동기식 응답 버스를 이용하여 캐쉬 코히어런스를 유지하는 한편 성능을 극대화한다. 정확한 수의 파이프 사이클을 판단하여 원격 사이드상의 캐쉬 상태를 시험하는 요구에 앞서 동작을 완료하는 것이 불가능하기 때문에, 비동기식 응답 버스는 모든 최종 응답에 대해 사용된다. 이들 응답은 원격 동작의 공식적 종료점을 표시하는 것으로서 이는 오리지널 요구기로 전송되기도 한다. 또한, 최종 응답에는 데이터 변형 정보(라인 변경 상태)가 붙여지며, 이것에 의해 로컬 디렉토리가 정확한 최종 상태로 갱신될 수 있게 된다. 최종 응답은 신호 최종 응답 인터페이스 버스에 대한 회선 쟁탈(contention)로 인해서 지연될 수 있으므로, 성능은 초기의 원격 캐쉬 교차 질의(XI)의 결과를 알게 되면 곧바로 보고하는 것에 의해 향상될 수 있다. RSC는 동기식 XI 응답 버스를 사용하여 최소 시간으로 그 XI 결과를 항상 보고할 수 있는데, 이것이 가능한 이유는 SC의 중앙 우선순위 스테이션이 원격 사이드로부터 수신된 어떤 새로운 요구도 적당한 파이프라인내로 즉시 들어가게 할 것이기 때문이다. 초기의 디렉토리 룩-업(look-up)은 일정한 파이프 사이클로 수행되며, 그 결과들은 XI 응답 버스를 통해 되전송된다. XI 응답으로서 알려진 동작을 개시한 인터페이스 제어기는 일정한 수의 장래의 사이클에서 이용될 수 있을 것이므로, 요구하고 있는 인출/저장 제어기에 적중/적중실패 결과를 전송하는데 사용된다.

제어 인터페이스의 관리에 부가하여, RSC IC는 모든 클러스터-클러스터 데이터 흐름을 또한 관리한다. 본 발

명은 각 파이프가 양 방향으로 16-바이트/사이클을 전송할 수 있을 정도의 충분한 데이터 흐름 I/O를 포함하며, RSC I/O는 잠재적인 16-바이트/사이클 최대 처리 능력을 이용할 수 있다. 두개의 단방향성 데이터 경로들이 있지만, 주어진 데이터 경로는 클러스터 A로부터 클러스터 B로 가며 또한 클러스터 B로부터 클러스터 A로 복귀하는 데이터를 저장하기 위한 통로로서의 역할을 해야 한다. RSC IC는 충돌을 방지할 뿐만 아니라 그를 행함에 있어 버스를 미리 확보해 두는 일이 없다. 따라서, RSC IC는 로컬 저장 제어기로부터의 요구들을 상주하는 원격 인출 제어기로부터의 요구들과 비교하여 인출 데이터를 복귀시키려 한다. 그들 요구가 데이터 경로를 차지하기 위해 서로 쟁탈을 벌이는 사이클 동안, 우선권은 복귀하는 인출 데이터에 주어진다. 따라서, 전체적인 성능이 향상된다. 또한, 인출 데이터가 원격 메인 메모리로부터 획득되는 경우, RSC IC는 대응하는 데이터 경로를 메모리 뱅크로부터의 데이터 액세스 시에 감시한다. RSC 데이터 경로를 이용할 수 있는 경우, 데이터는 원격 인출 버퍼를 우회할 것이며, 따라서, 데이터의 임시적인 버퍼링과 연관된 통상의 대기 시간이 감소된다.

본 발명에 따르면, 전체적인 시스템 처리능력의 향상을 위해 복제된 원격 저장 제어기 자원의 관리를 개량한 방법이 제공된다. 예를 들어, 원격 캐쉬에 대한 적중을 발생하는 일련의 연속적인 인출 요구들의 효율을 최대화하는데 사용되는 한가지 기법은 복제 원격 인출 자원들에 작업 요구들을 교번적으로 사용하고자 하는 것이다. 모든 인출은 전송되고 있는 제 1의 16-바이트 데이터와 일치하는 최종 응답으로 종료된다. 이 방법에 의하면, 원격 인출 자원이 즉시 새로운 작업에 할당될 수 있으며, 원격 인출 제어기의 버퍼는 여전히 공백 상태로 유지되는 장점이 제공된다. 단점은 새로운 작업이 원격 캐쉬에 대해 적중을 발생하는 데이터 인출인 경우에 나타난다. 버퍼는 여전히 공백 상태로 유지되므로, 원격 인출 제어기는 버퍼가 이용가능하게 될 때까지 파이프를 통해 요구를 재순환시켜야 한다. RSC IC는 그러한 시나리오를 이용가능한 복제 원격 인출 제어기에 제 2의 인출 요구를 보내는 것에 의해 배제시킨다. 따라서, 제 2의 원격 인출 제어기는 그의 버퍼에 대한 로딩을 개시할 수 있는 한편 제 1 버퍼는 그의 데이터 전송을 완료한다. 따라서, 제 1 버퍼의 전송 완료시에 제 2 버퍼는 그의 데이터를 즉시 인터페이스를 통해 전송할 수 있다.

원격 인터페이스는 동작이 원격 사이드상에서 성공적으로 완료될 수 없는 경우에 많은 동작들이 검색될 수 있게 하는 것에 의해 높은 수준의 신뢰성 및 가용성을 또한 나타낸다. 이들 유형의 문제점은 원격 거절(remote reject) 및 인터페이스 에러와 같은 두가지의 주요 카테고리로 분류된다. 원격 클러스터는 잠재적인 크로스-클러스터 데드록(cross-cluster deadlock)을 피하기 동작을 거절할 수도 있다. 이러한 데드록은 저장 서브시스템 인출 및 저장 제어기들이 그들에 대해 서비스하는 RSC 자원보다 많은 경우에 발생될 수 있다. 원격 인출 및 저장 제어기는 크로스-클러스터 데드록을 유발할 수 있는 동작 시퀀스를 감시하도록 구성된 데드록 회피 메카니즘을 구비한다. 이러한 시나리오의 검출시, 그 제어기는 특정 거절 응답을 개시(initiating) 클러스터에 되돌려 보냄으로써 현안중의 동작(pending operation)을 거절할 것이다. RSC IC는 그 거절을 발신(originating) 인출/저장 제어기로 보낼 것이므로 동작이 재시도될 수 있다. 동작들은 데드록 윈도우가 사라질 때까지 연속적으로 거절되고 재시도될 수 있다. 다른 유형의 복구(recovery)는 인터페이스 패리티 에러가 새로운 RSC 동작을 수반하는 어떤 제어 정보상에서 검출되는 경우에 발생한다. 동기식 인터페이스는 명령 전송후 일정 수의 사이클내에서 인터페이스 에러 상태를 전송하는데 사용된다. 에러가 있는 경우, 발신 인출/저장 제어기는 그를 통보받고 복구의 적격성을 판단한다. RSC IC는 대응하는 RSC 자원을 자동적으로 리세트시켜 동작이 다시 요구될 수 있게 한다. 이들 및 다른 개선점에 대해서는 다음의 상세 설명을 통해 개시하겠다. 본 발명과 그의 장점 및 특징에 대해서는 도면을 참조한 다음의 설명으로부터 더욱 잘 이해될 것이다.

본 발명을 그의 양호한 실시예에 대해 설명하나, 당업자라면, 본 명세서에 개시된 개념을 3개 이상의 클러스터를 구비하며 양호한 실시예의 것과는 다른 저장 클러스터를 이용하는 시스템들에 적용될 수 있음을 알 것이다. 또한, 본 발명에서는, 이들에 국한하고자 하는 것은 아니나 캐쉬 구조, 메인 메모리 구성, 데이터 경로 자원들의 수 및 크기, 각종 제어기의 구성 및 파이프라인들의 수 및 크기를 비롯해서 기능 유니트들의 개수 및 구성을 달리한 변형된 저장 제어기 실시예들이 생각될 수 있을 것이다.

발명의 구성 및 작용

도 1a에는 2-노드 대칭적 멀티프로세서 시스템의 저장 제어기 클러스터를 도시한다. 본 발명은 동일한 로컬 인출/저장 제어기(Local Fetch/Store Controller)(11) 세트, 원격 인출/저장 제어기(Remote Fetch/Store Controller)(12) 세트 및 중앙집중식 멀티스테이지 파이프라인(Centralized Multistage Pipeline)(13) 세트와 상호작용하는 원격 저장 클러스터 인터페이스 제어기(Remote Storage Cluster Interface Controller: RSC IC)(10)로서 제시된다. 이들 파이프라인은 중앙 우선순위 스테이션(Central Priority Station)(14)으로부터 신호를 공급받는다. 중앙 우선순위 스테이션(14)은 N개의 중앙 처리 유니트 제어기(Central Processing Unit(CPU) Controller)(15) 또는 N개의 I/O 아답터 제어기(Adapter Controller)(16)로부터의 비동기 작업 요구들을 우선순위화한다. 각각의 CPU 제어기는 SMP 시스템의 특정한 CPU에 적합하다. 파이프라인들은 레벨 2 스토어-인 캐쉬(Level 2 store-in cache)를 공유한다. 이 레벨 2 스토어-인 캐쉬에 대해서는 모든 중앙 처리 유니트, I/O 아답터 및 로컬 인출/저장 제어기 및 원격 인출/저장 제어기에 의한 판독 및 기록 액세스가 수행된다. 양호한 실시예에서 이용하는 캐쉬 코히어런시(coherency)에 의하면, 데이터가 캐쉬내에, 두 클러스터상의 모든 프로세서 및 I/O 아답터가 데이터의 카피를 액세스할 수 있게 하는 판독 전용 상태로 또는 단일 프로세서가 어떤 시점에서도 데이터를 소유할 수 있게 하는 독점권 상태(exclusive ownership state)로 상주할 수 있다. 어떤 클러스터상의 어떤 프로세서는 다른 프로세서가 데이터를 현재 소유하고 있다고 하더라도 어느 시점에서도 그 데이터의 소유권을 요구할 수 있다.

양호한 실시예에서는 이중 파이프라인 구성을 이용하는데, 이 구성에서는 중앙 파이프라인(13), 원격 인출/저장 제어기(12) 및 로컬 인출/저장 제어기(11)가 모두 복제된다. 본 발명에서 개시하는 개념들은 저장 제어기(SC)의 주변 구조와는 관계없는 것으로서, 이들 개념은 3개 이상의 파이프라인을 이용하는 덜 전통적인 단일 파이프라인 SC 구성 또는 더욱 전통적인 SC 구성으로 효과적으로 구현될 수 있다. 당업자라면, RSC IC(10) 인터페이스 및 내부 기능 블럭이 대부분의 어떤 SC 구조에도 적합하도록 용이하게 치수화할 수 있다.

RSC 인터페이스 제어기(10)는 각종 SC 기능 유니트와 상호작용하는 수개의 서브-유니트로 구성된다. 동일한 파이프라인(13) 세트 및 인출/저장 제어기(11, 12) 세트로부터의 요구들에 대한 서비스를 행하는데 단일의 원격 제어 클러스터만이 사용되므로, 그 단일의 RSC IC는 다수의 로컬 인터페이스 및 클러스터-클러스터 인터페이스를 수용할 수 있어야 한다. 또한, 그 단일의 RSC IC는 로컬 클러스터로부터 원격 클러스터로의 통신량 및 원격 클러스터로부터의 통신량을 관리해야 한다. 도 1b는 RSC IC(10)를 구성하는 서브-유니트들 및 인터페이스들을 상세히 도시한 것이다.

양호한 실시예에서 사용되는 저장 제어기의 특성 때문에, 대부분의 요구들은 로컬 인출/저장 제어기(11)로부터 나온다. 이들 요구는 LFAR 요구로서 알려진 인출 요구들로 분할되며, LSAR 요구로서 알려진 요구들을 저장한다. 양호한 실시예에서는, 각 파이프라인에 대해 4개의 LFAR 및 4개의 LSAR 요구기가 존재하므로, 전체 16개의 가능한 LFSAR 요구기들이 RSC 인터페이스를 사용하기 위해 쟁탈전을 벌린다. LFSAR 제어기들내의 사전 우선순위 스테이션은 각 파이프로부터 하나의 요구기를 선택하여 임의의 한개 클럭 사이클에서 최대 두개의 요구를 전송한다. 본 발명에 의하면, 소정 수의 LFAR 및 LSAR 요구기들이 소정 수의 파이프라인간에 분배될 수 있게 한다. 또한, LFSAR 제어기(11)들내의 사전 우선순위 스테이션은 RSC IC내의 명령 우선순위내에 직접 구성될 수 있다.

LFSAR 제어기(11)들과 RSC IC(10)간의 인터페이스를 참조하면, 파이프라인과 연관된 각각의 제어기 세트에 대해 한개의 인터페이스가 존재한다. 각각의 인터페이스는 요구 버스, C3 명령 버스, C3 요구기 ID 버스 및 C3 어드레스 버스로 이루어진다. 8-비트 요구 버스는 각각의 LFAR 및 LSAR 요구기에 대해 한개의 비트를 포함하는데, 이들 중의 한개의 비트만이 소정의 사이클에서 활성화될 수 있다. 이 비트는 그 사이클에서 사전 우선순위 스테이션이 선택한 LFAR 및 LSAR 제어기의 ID를 나타낸다. 이에 대응하는 버스들은 명령을 파이프라인내로 발생하기 시작한 CPU, I/O 아답터 또는 SC 제어기의 명령, 어드레스 및 ID를 전달한다. 이 정보의 모두는 파이프라인의 제 3 스테이지에 대응하는 C3 사이클에서 RSC IC에 제시된다. 정보가 제시되는 사이클에서 LFSAR 요구에 응하도록 RSC IC를 사용할 수 없는 경우에는, LFSAR 제어기들은 다음 사이클에서 계속 동일한 요구를 제시할 수도 있거나 새로운 요구를 동적으로 선택할 수도 있다.

LFSAR 제어기외에도, 파이프라인(13) 자체도 "신속-경로화(fast-pathing)"라고 일컬어지는 동작을 가능하도록 하는 요구기로서의 역할을 한다. 이 신속-경로화 동작은 RSC가 두 파이프 모두를 감시하여 원격 인출 동작이 요구되고 어떤 LFSAR 제어기(11)로부터의 어떤 작업도 현안중에 있지 않는 경우에 그 원격 인출 동작에 착수한다. C1 명령, C1 어드레스 및 C1 요구 ID는 각 파이프의 제 1 스테이지(C1 사이클)로부터 획득되며 RSC IC의 RSC 우선순위(21) 서브-유니트내에 있는 사전 우선순위 스테이션으로 보낸다. 사전 우선순위 스테이션의 출력은 RSC 우선순위(21) 서브-유니트내에 또한 배치된 메인 우선순위 스테이션으로 보내지며, 여기서 LFSAR 제어기(11)로부터의 요구와 쟁탈전을 벌린다.

매 사이클에서, RSC 우선순위(21) 스테이션은 현안중의 작업 요구를 시험하고 고성능 동작을 사용해서 어떤 요구가 인터페이스를 통과하도록 해야 하는 지의 여부를 결정한다. 하나의 요구가 선택되면, 선택된 동작의 파이프라인에 대응하는 LFSAR 제어기(11)에 승인(grant) 신호가 보내진다. 승인 신호는 선택된 동작이 LFSAR 제어기(11)로부터의 현안중의 요구이었는 지의 여부 또는 신속-경로화 동작이 파이프라인(13)으로부터 개시되었는 지의 여부를 나타낸다. 승인 신호가 LFSAR 제어기로 보내지고 있는 동안, 선택된 명령 및 그와 연관된 RSC를 나타내는 어드레스, 요구기 ID 및 태그 라인(tag line)이 RSC 인터페이스를 통해 원격 클러스터로 보내진다.

모든 RSC 동작에는 원격 클러스터로부터의 몇가지 유형의 완료 응답(completion response)이 필요하다. 또한, 데이터 인출에는 요구된 데이터가 원격 캐쉬내에 존재하는 지의 여부를 나타내는 교차 질의(Cross Interrogate)(XI)가 필요하다. 모든 응답은 RSC IC(10)를 통해 처리되며 직접적으로 또는 간접적으로 오리지널 요구기(original requester)로 보내진다. 응답이 디코딩되는 대부분의 시간 및 적정한 상태(status), 해제(release) 및 캐쉬 코히어런시 정보는 LFAR 또는 LSAR 제어기에 보내진다. 그러나, 오리지널 CPU 제어기(CFAR 15)에 응답을 되돌려 보내는 동작이 많다. 응답의 최종 목적지에 관계없이, RSC IC는 모든 RSC 동작에 대한 응답 정보가 단일의 인코디드 응답(Encoded Response) 버스에서 다중화될 수 있도록 매 동작에 대해 필요한 모든 정보를 추적한다. RSC IC는 응답 처리기(Response Handler)(22)내로 입력되는 응답을 수신한다. 이 응답 처리기의 업무는 응답을 디코딩해서 적당한 정보를 LFSAR 제어기(11) 또는 CPU 제어기(15)로 송신한다.

도 1b는 원격 인출/저장 제어기(12)와 RSC IC간의 인터페이스를 또한 도시한다. 원격 인출/저장 제어기(12)는 원격 인출(RFAR 12a) 및 원격 저장(RSAR 12b)으로 분할된다. 원격 인출 제어기(12a)는 다른 클러스터로부터의 인출 요구들을 수신하고 그들을 파이프라인(13)을 통해 처리하며 (가능하다면) 데이터를 필요한 응답 정보와 함께 복귀시키는 역할을 담당한다. 원격 저장 제어기(12b)는 입력 저장 동작(및 어떤 수반되는 데이터)을 수신하고 그들을 파이프라인(13)을 통해 처리하며 필요한 응답 정보를 복귀시키는 역할을 담당한다. 각각의 파이프라인은 그와 연관된 RFAR(12a) 및 RSAR(12b) 제어기를 가지므로, 최대 4개의 요구가 RSC IC에 제시되어 주어진 사이클에서 인코디드 응답 버스에 정보를 복귀시킬 수 있다. RSC IC내의 응답 우선순위(23) 서브-유니트는 그들 요구를 중재하며 인코디드 응답 버스의 통신량을 관리한다. RFAR 제어기가 인출 데이터를 복귀시켜야 하는 경우, 응답 우선순위(23) 스테이션은 RSC 우선순위(21) 및 XPT 제어기(25)와 통신을 행하여 데이터 경로가 이용가능하도록 한다.

본 발명의 주요 관점들 중의 하나는 자원 레지스터(Resource Resister)(24)를 사용해서 로컬 사이드의 원격 활동(activity)을 추적하는 것이다. 인터페이스 I/O를 최소화하고 처리 능력을 극대화하기 위해, RSC IC는 로컬 사이드의 오리지널 요구기에 대한 대리 프로세스로서 작용한다. 이것은 원격 사이드의 RFAR 및 RSAR 자원을 추적하여 일정한 클러스터-클러스터 통신의 필요성을 배제시킨다. 자원 레지스터(24)는 RSC 우선순위(21) 스테이션과 상호작용하여 RSC 자원이 이용가능한 경우에만 동작이 개시되도록 한다. 동작이 개시되면, RSC IC는 선택된 RFAR 및 RSAR을 "사용중(in use)"으로서 표시하고, 그 자원은 동작 완료를 나타내는 응답이 수신될 때까지 사용중의 상태로 유지된다. 이들 응답은 그다음 자원을 리세트시켜 새로운 동작에 이용가능하게 되도록 하는데 사용된다.

양호한 실시예에서는, 각 파이프라인에 대해 2개의 RFAR 및 2개의 RSAR로 이루어진 총 8개의 RSC 자원 레

지스터(24)가 사용된다. 두 RFAR/RSAR의 각각은 서로 동일하며, 주로 각 파이프가 다수의 원격 인출 및 저장 동작들을 동시에 처리될 수 있게 하는 것에 의해 성능을 향상시키기 위해 존재한다. 다시 주목할 것은 본 발명에서는 각 파이프마다 2개의 RFAR 및 2개의 RSAR가 필요치 않으며 또한 이들 RFAR/RSAR에도 국한되지 않는다는 것이다. 모든 원격 자원은 그의 수에 관계없이 동일한 방식으로 추적된다.

동작을 선택하기 전에, 오리지널 파이프라인 명령은 RSC 명령으로 변환된다. 대다수의 경우에서는 결과적인 RSC 명령이 오리지널 명령과 동일하지만, 어떤 경우에는 명령 코드 포인트가 재맵핑되어 유사한 동작들이 단일의 RSC 코드 포인트를 공유할 수 있게 한다. 이 단계는 또한 모든 RSC 인출 명령이 연속 영역(양호한 실시예의 '01'x-'1F'x)내에 있게 하며 모든 저장 명령이 다른 연속 영역(양호한 실시예의 '20'x-'3F'x)내에 있게 한다. 동작이 개시되면, RSC IC는 2개의 선택 비트를 사용하여 8개 자원중의 어떤 것이 새로이 선택된 동작에 대한 서비스를 행해야 하는 지를 다른 사이드에 지시한다. 그들 두 비트는 파이프라인을 참조하며, 그 파이프라인내에 있는 동일 자원중의 어떤 것이 명령을 처리할 것이다. 명령의 비트 0은 그 명령이 인출 타입(비트 0 = 0)인지 또는 저장 타입(비트 0 = 1)인지를 판단한다. 모든 인출 명령은 RFAR이 서비스하며 저장 명령은 RSAR이 처리한다. 명령 및 어드레스는 항상 전송되므로, 이 방식에서는 두 사이드를 동기시키기 위해 단지 한 번만 전송되는 2개의 부가적인 인터페이스 제어 비트만이 필요하다. 또한, 주목해야 할 것은 양호한 실시예에서 명령, 어드레스 및 선택 라인외에 요구 ID를 전송하기 위한 RSC 인터페이스 버스를 개시한다는 것이다. 이 요구 ID는 RSC 인터페이스를 통해 다른 원격 사이드의 CPU 제어기 또는 I/O 아답터 제어기와 같은 요구기에 전송되는 순수한 데이터이다. 본 발명이 이용하는 기법들에서는, 특허청구범위에 개시한 목적들을 달성하기 위해 오리지널 요구기의 ID를 하등 알아야 필요가 없다.

최종 서브-유니트는 크로스포인트(XPT) 제어기(25)로서, 이 제어기는 클러스터들을 연결하는 4개의 데이터 경로를 관리하는 역할을 담당한다. 양호한 실시예에서는, 각 파이프에 대해 2개의 단방향성 데이터 경로가 존재하므로, 4개의 데이터 전달이 동시적으로 발생할 수 있다. 각 데이터 경로는 자신의 XPT 버스를 가져 4개의 동작이 동시적으로 발생할 수 있게 한다. 데이터 경로는 16-바이트의 폭을 가지며, 매 사이클에서 쿼드워드(quadword)(16-바이트)를 전달한다.

본 발명의 목적들 중의 하나는 원격 자원 관리를 이용해서 원격 인출/저장 제어기(12)의 크기 및 복잡성과 클러스터-클러스터 인터페이스를 통해 교환되어야 하는 정보의 양을 최소화하고자 하는 것이다. 복잡한 저장 제어기들을 가진 하이-엔드(high-end) SMP 시스템에서는, 사실상 로컬 클러스터내에서 개시될 수 있는 어떤 명령도 원격 클러스터에서의 처리를 위해 인터페이스를 통해 또한 전달될 수 있다. 이들 명령을 일련의 단위 동작으로 분해하면, 원격 사이드의 RSC 인출/저장 제어기(12)가 동일한 상태 기계(status machine)를 사용하여 수개의 유사한 명령을 처리할 수 있음을 알 수 있다. 그러므로, RSC 구성을 간단하게 하기 위해, 로컬 사이드의 어떤 오리지널 명령들이 등가의 "베이스(base) "RSC 명령으로 재맵핑된다. 예를 들어, "저장 보호키에 의한 인출 독점권(fetch exclusive with storage protect key)"은 파이프라인 시퀀스 및 디렉토리 갱신 조치가 "키에 의하지 않는 인출 독점권(fetch exclusive without key)"과 동일하게 되도록 한다. 그러므로, RSC 인터페이스 제어기는 키에 의한 인출 독점적 명령('06'x)을 인터페이스를 통해 전송하기 전에 그 명령을 간단한 독점적 명령('02'x)으로 재맵핑할 것이다.

양호한 실시예는 도 2에 도시한 바와 같이 하드웨어로 명령 변환을 구현한다. 오리지널 명령은 명령 변환 테이블(28)의 플립 비트(Flip bit)를 구현하는데 필요한 로직 게이트들로 구성된 플립 비트 발생기(26)내로 들어간다. 오리지널 명령은 디렉토리 상태 및 타겟 L3과 조합되어 어떤 비트들(있는 경우) 플립될 필요가 있는 지를 판단한다. 결과의 플립 비트들은 XLAT(27) 블록에서 오리지널 명령과 배타적 오어(OR) 연산되어 명령 변환 테이블(28)에 도시된 원하는 RSC 베이스 명령으로서 발생된다.

RSC 명령 변환기는 단일 클럭 사이클에서 변환을 수행하도록 구성되며, 독립적인 기능 유니트이다. 그러므로, 당업자라면, 그 변환기가 유연하게 이용될 수 있음을 알 수 있을 것이다. 예를 들어, 그 명령 변환기는 RSC 인터페이스 제어기(10)의 일부로서 물리적으로 구현될 수 있고 또는 RSC에 대한 작업 요구를 개시하는 제어기내에 포함될 수도 있다. 또한, 그 명령 변환기는 RSC 명령 우선순위 스테이션과 동일한 로직 사이클내에 통합될 수도 있고 또는 오리지널 명령이 이용가능한 경우에 이전 사이클에서 수행될 수도 있다. 예를 들어, 양호한 실시예에서, 파이프라인 신속-경로 명령은 제 2의 파이프라인 스테이지(C2)에서 이용가능하므로, 그 명령은 RSC 명령 우선순위 사이클 전에 변환될 수 있다.

명령 변환을 사용하면, 인터페이스 효율이 여러 가지 방법으로 향상된다. 첫째, 대부분의 동작에서는 원하는 데이터가 로컬 캐쉬내에 있는 경우 원격 사이드를 질의를 할 필요가 없다. 그러므로, 명령 변환기는 디렉토리 상태를 사용하여 그들 유형의 동작이 RSC 인터페이스를 사용하지 않게 할 것이다. 둘째, 명령 변환에 의하면, 데이터 전달에는 일단 전달되고 처리된 다음에 오리지널 사이드로 다시 복귀되는 것이 아니라 단지 한 방향으로 전송되는 것만이 필요하게 된다. 예를 들어, 변환되지 않은 I/O 저장 64-바이트(명령 28)는 데이터의 최종 목적지가 로컬 L3 메모리인 경우에도 64-바이트의 데이터가 무조건적으로 전송되게 한다. 이는 64-바이트가 원격 사이드로 전달되어 타겟 라인내로 합병될 것이고 그 갱신된 라인이 인터페이스를 통해 다시 되돌아 와서 로컬 클러스터에 부착된 L3 메모리내에 저장될 수 있게 할 것임을 의미한다. 본 발명은 로컬 L3 및 디렉토리 상태를 이용하여 목적지 어드레스가 원격 L3 메모리이고 데이터가 캐쉬내에 상주하지 않는 경우에 인터페이스를 통해 64-바이트만을 전송하도록 하는 것에 의해 데이터 전달을 최적화한다. 캐쉬에 대한 데이터 적중실패가 발생하면, 원격 캐쉬에서의 데이터 적중이 발생하는 경우 그 원격 캐쉬로부터의 데이터 전달을 요구하는 다른 사이드에 질의가 전송된다. 이러한 시나리오에서는, 어떠한 초기 데이터 전달도 발생하지 않으며, 데이터는 단지 타겟 라인이 원격 캐쉬에 보유되고 있는 경우에 인터페이스를 통해 복귀할 것이다. 데이터가 원격 캐쉬내에 있는 경우에도, 전체적인 동작에서는 단지 원격 사이드로부터 로컬 사이드로의 단일 데이터 전달만이 필요하며, 로컬 사이드에서 그 데이터는 I/O 저장 데이터와 합병되고 로컬 L3 메모리에 저장될 수 있다. 마지막으로, I/O 저장에 대한 제 3의 가능한 시나리오는 타겟 데이터가 두 캐쉬내에 판독 전용 상태로 상주하는 경우이다. 이 경우, I/O 저장 데이터는 로컬 데이터 카피와 합병되어 어떤 데이터도 인터페이스를 통해 전달될 필요가 없게 할 수 있다. 그 대신, 오리지널 명령이 판독 전용 무효화 명령으로 변환된다. 이 판독 전용 무효화 명령은 원격 인출 제어기(12a)로 보내져 원격 캐쉬내의 데이터 카피가 무효로서 표시될 수 있

게 한다.

작업 요구를 가능한 효율적이고도 촉진적으로 처리하기 위해서, RSC 인터페이스 제어기(10)는 다중 레벨의 인공 지능적 우선순위 스테이션을 이용한다. 도 3a는 메인 명령 우선순위 스테이션(33)과 이것에 신호를 공급하는 파이프라인 사전 우선순위 스테이션(32)을 구비한 가진 전체적인 우선순위 스테이션을 도시한 것이다. 파이프라인 사전 우선순위 스테이션(32)은 신속-경로화 후보를 찾는 두 파이프라인중의 제 1 스테이지(C1)를 감시한다. 어떤 CPU 인출 명령('01'x-'07'x) 명령은 신속-경로화의 후보로서 고려된다. 어느 한쪽의 파이프라인 명령이 후보인 경우, 그 명령은 사전 우선순위 스테이션(32)내로 들어가 C2 스테이징 영역내로의 선택을 위해 다른 쪽의 파이프라인 명령과 쟁탈전을 벌린다. 단지 하나의 파이프가 그 사이클에서 유효한 후보를 가지는 경우에는 그 것이 자동적으로 선택될 것이다. 한편, 양쪽 파이프가 유효 후보를 가지는 경우에는 간단한 라운드 로빈 방법에 의해 순번이 판단된다.

파이프 명령이 C2 스테이징 영역내로 선택될 때마다, 그 명령은 제 2의 파이프 스테이지(C2)와 연관된 각종 인터페이스 신호와 비교된다. 이들 C2 거절 신호는 디렉토리 상태, 각종 CPU 제어기(15)로부터의 거절 신호 및 LFSAR 제어기(11)로부터의 차단(block) 신속-경로 신호로 이루어진다. 이들 신호의 조합은 C2 스테이징 영역내의 현재 동작을 완전히 거절해야만 하는지 또는 메인 명령 우선순위 스테이션(33)으로 보내야 하는지의 여부를 판단한다. 동작에 대해 있을 수 있는 거절 이유로서는 다음과 같은 것들이 있다.

* 원격 사이드에 대한 데이터 질의의 필요성을 부정하는, 적정 상태로 로컬 디렉토리에 대한 적중을 발생하는 CPU 인출.

* CPU CFAR 제어기(15)들중 어떤 것으로부터의 거절 신호.

* 리세트 상태의 C2 파이프라인 유효.

* LFSAR 제어기(11)로부터의 차단 신속-경로 신호.

* L3 메모리 구성 어레이로부터의 무효 어드레스 표시.

거절 조건중의 어느 것도 나타나지 않는 경우, 명령은 메인 명령 우선순위 스테이션(33)에 보내지며 이 스테이션에서 그 명령은 두 LFSAR 제어기(11)로부터의 요구와 쟁탈전을 벌린다.

도 3a에 도시한 바와 같이, 명령 우선순위 스테이션(33)은 각 LFSAR 제어기(11)로부터의 신호 세트와 파이프라인 사전 우선순위 스테이션(32)으로부터 제공되는 파이프라인 속성 경로 정보를 수신한다. 또한, 명령 우선순위 스테이션(33)은 자원 레지스터(24) 및 XPT 제어기(25)와 또한 인터페이스를 이루어 인공 지능적으로 적당한 동작을 선택할 수 있게 된다.

기본적으로, 그 동작은 LFSAR 동작이 현안중의 것이고 RSC 자원이 이용가능한 경우에는 LFSAR 동작을 항상 선택하려고 할 것이다. 단일의 LFSAR 제어기(11)만이 요구하고 있고 RSC 자원이 이용가능한 경우에는 그 자원이 선택된다. 둘 모두의 LFSAR 제어기(11)가 요구하고 있고 하나의 LFSAR 제어기만이 이용가능한 자원들을 갖는 경우에는 그 제어기가 쟁취할 것이다. 둘 모두의 LFSAR 제어기(11)가 요구하고 있고 둘 모두가 이용가능한 자원을 갖는 경우에는 인출 유형의 동작이 저장 유형의 동작에 우선할 것이다. 둘 모두의 요구가 동일한 유형의 것인 경우에는 간단한 라운드 로빈 방법에 의해 순번이 판단된다. 마지막으로, 어떠한 LFSAR 제어기(11)도 요구하고 있지 않거나 어떠한 자원도 LFSAR 요구에 응하도록 이용가능하지 않은 경우에는 신속-경로 요구가 선택된다.

자원의 가용성은 동작의 유형에 따른다. 인출이 가장 간단한 경우인데 그 이유는 필요한 자원만이 인출 동작을 처리하는 파이프라인에 대응하는 원격 인출 제어기(RFAR 12a)이기 때문이다. RFAR은 이용가능하지 않을 수도 있는데 그 이유는 그들 RFAR이 다른 인출 동작을 처리하기에 바쁘거나 자원 디스에이블 스위치(resource Disable switch)가 활성상태일 수도 있기 때문이다. 명령 우선순위 스테이션(33)은 각각의 RSC의 디스에이블 스위치 및 유효 비트를 관리하여 가용성을 판단한다.

시스템 성능을 더욱 향상시키기 위해서, 우선순위 동작은 자원 처리기(22)와 연관하여 작업을 수행해서 원격 인출의 효율성을 극대화한다. 통상, 인출 요구는 다음의 이용가능한 RFAR 제어기(12a)에 디스패칭될 것이다. 원격 클러스터의 RFAR은 파이프라인내의 인출 요구를 처리하고 그의 데이터 버퍼에 대한 로딩을 시작한다. 이와 동시에, 그 RFAR은 원격 사이드의 RSC IC(10)에 대해 요구를 발생해서 최종 응답 및 데이터를 인터페이스를 통해 복귀시킨다. 최종 응답이 전송되는 즉시, 그 RFAR 자원은 이용가능한 것으로서 고려되며 새로운 작업을 받아들일 수 있다. 새로운 비-데이터 동작이 그 RFAR에 전송되면, 그 RFAR은 그 동작을 처리하는 한편 이전 인출로부터의 후행(trailing) 바이트들은 여전히 데이터 버퍼로부터 판독된다. 그러나, 새로운 동작이 제 2의 데이터 인출인 경우, 그것은 버퍼가 이용가능하게 될 때까지 계속해서 원격 사이드의 파이프라인을 통해 재순환될 것이다.

이러한 시나리오는 제 2의 인출 요구가 도달할 시에 둘 모두의 자원이 이용가능한 경우 연속 데이터 인출이 교번적 RFAR 제어기에 전송되도록 하는 것에 의해 본 발명에 포함된다. 예를 들어, RFAR A0이 제 1 인출을 처리하고 있고 제 2 인출이 RFAR A0의 사용중에 도달하면, 제 2 인출은 RFAR A1(RFAR A1이 이용가능한 것으로 가정함)로 라우팅될 것이다. 또한, RFAR A0이 최종 응답이 전송되고 있기 때문에 이용가능하게 되고 제 2 인출이 도달하면, 제 2 인출은 또한 RFAR A1로 라우팅될 것이다(이는 RFAR A0 버퍼가 여전히 데이터를 전송하고 있기 때문이다.). 그러나, 판독 전용 무효화와 같이 비-데이터 동작이 RFAR A0의 사용중에 도달하는 경우에는, 그 동작은 RFAR A1로 라우팅될 것이다. 이 동작 다음에 제 3의 데이터 인출 동작이 뒤따르고 RFAR A0이 이용가능한 경우에는, 그 새로운 데이터 인출은 버퍼가 여전히 후행 바이트를 전송하고 있더라도 RFAR A0에 전송될 것이다. 달리 말해서, 요구들을 다른 RFAR들에 번갈아 사용하기 위한 메카니즘은 한 쌍의 자원중 어떤 자원에 대한 가용성에 도움을 준다.

도 3b는 상기한 메카니즘과 RSC IC 우선순위 스테이션(21)간의 상호작용을 보여주기 위한 로직 블록도이다. 자원 토글링 기능(Resource Toggling function)은 각 RSC 자원(24) 쌍의 RSC IC내에 존재한다. 파이프 A에 대한 RFAR 쌍을 나타내는 단일 자원 토글러(35)는 도 3b에 도시된다. 이 토글러는 각 RFAR 자원(A0 및 A1)으로부터의 가용성 신호를 수신한다. 이들 가용성 신호는 6개 다른 RSC 자원으로부터의 가용성 신호와

함께 RSC 우선순위 스테이션(21)에도 공급된다. 또한, RSC우선순위 스테이션이 발생한 파이프 A 인출 승인 신호도 자원 토글러(35)에 공급된다. 마지막으로, 자원 토글러는 토글 보유 래치(Toggle Possession Latch)(36)를 사용하여 자원 쌍중의 어떤 자원을 다음의 동작에 할당할 것인지를 제어한다(그 권리 조건이 나타나는 경우). 자원 토글러는 단일 선택 신호를 생성하는데, 이 신호는 인출 승인 신호와 2회에 걸쳐 AND 연산되어 로드 RFAR A0 및 로드 RFAR A1 신호들로서 발생된다. 이들 로드 RFAR A0 및 로드 RFAR A1 신호는 RFAR A0 및 RFAR A1 자원 레지스터에 전송된다.

도 3b에 도시한 토글러 진리표(37)는 선택 신호 및 토글 보유 래치(36)가 어떤식으로 갱신되는지를 도시한 것이다. 두 자원중의 하나의 자원만이 이용가능한 경우, 선택 신호는 토글 보유 래치(36)의 상태와 무관하게 이용가능한 자원 신호에 대해 이행되지 않을 것이다. 둘 모두의 자원이 이용가능하고 인출 승인이 파이프라인에 발생된 경우, 토글 보유 래치의 현재 상태는 선택 신호를 구동시킨다. 게다가, 토글 래치는 '다른' 자원에 대해 조정되어야 하는 후속 인출을 기대하여 다음 사이클에서 갱신된다(이용가능한 경우). 토글 진리표(37)의 가장 아래쪽에 도시한 바와 같이, 이용가능한 신호는 자원 유효 비트(59a), 디스에이블 비트(59f) 및 RST_RFAR_A0 래치의 함수로서, 이 함수는 최종 응답이 그 자원에 대해 수신되었고 그 응답이 이 사이클에서 "이용가능한" 것으로서 고려됨을 나타낸다.

저장 유형의 동작은 인출보다 더 복잡한데, 이는 그들 동작이 그 명령의 수반을 위해 초기 데이터 전달을 포함할 수도 있기 때문이다. 명령 우선순위 스테이션(33)은 디스에이블 스위치 및 유효 비트를 테스트하는 것에 의해 RFAR과 동일한 방식으로 원격 저장 제어기(RSAR 12b)에 대한 가용성의 일부를 판단한다. 명령 우선순위 스테이션(33)은 RSC 명령을 디코딩하여 데이터 전달이 필요한지를 알아본다. 명령은 데이터 전달을 필요로 하지 않으면, 데이터 버스의 가용성이 테스트되어야 한다. 이러한 테스트를 통과하기 위해서는 다음과 같은 두가지의 조건이 충족되어야 한다.

1. 저장 유형의 동작을 발생하는 파이프라인에 대응하는 데이터 버스가 데이터를 전달하는데 사용되고 있지 않아야 한다.

2. 대응하는 RFAR 제어기(12a)가 데이터 경로를 사용하여 다른 클러스터로부터 발생된 인출 동작을 위한 데이터를 복귀시킬 것을 요구하고 있지 않아야 한다.

이들 두 조건이 충족되고 판독 저장키 동작(명령 30)과 같은 저장 명령이 데이터 경로의 사용을 필요로 하지 않으면, 저장 동작에 대한 자원 기준이 충족된다.

상기한 우선순위 동작은 자원이 이용가능하면 새로운 동작이 원격 사이드로 전송되도록 할 것이다. 게다가, LFSAR 제어기(11)를 이용하는 것에 의해, LFSAR 자원의 처리 능력이 향상됨으로써, 상호 연동된 상태로 다른 자원의 사용 완료를 기다리는 자원들로 인해서 유발되는 폭주 현상 및 데드록 현상이 줄어든다. 일단 동작이 선택되면, (오리지널 또는 재맵핑된 코드포인트내의) 명령은 전체 27-비트 어드레스의 RSC 인터페이스를 통해 전송된다. 원격 사이드의 RSC 명령 분배기는 그 명령을 c0_c1_cmd 명령 버스의 값에 따라 RFAR 및 RSAR로 라우팅한다. 또한, c0_c1_pipe_sel 및 c0_c1_req_reg_sel은 그 동작을 처리하는 파이프라인에 관련하는 선택된 RFSAR 제어기(12a 및 12b)의 조정에 사용된다. 이 원격 자원 관리 기법은 다수의 파이프라인간에 분포된 다수의 저장 제어기 자원이 제한된 수의 I/O를 사용하는 공유 RSC 인터페이스를 이용할 수 있게 한다.

전술한 바와 같이, c0_c1_reqid는 인터페이스를 통해 전송되나 원격 관리 동작에 참가하지 않는 발신 요구기의 ID이다. 이 ID는 원격 사이드로 전송되고 그를 필요로 하는 저장 제어기를 따라 통과되는 정보로서 순수하게 처리된다.

새로운 동작의 발생시, 명령 우선순위 스테이션(33)은 승인 신호를 대응하는 LFSAR 제어기(11)에 발생한다. 양호한 실시예에서 임계적인 타이밍 경로는 이 승인 신호를 1 사이클만큼 지연시키는 것에 의해서 완화된다. 그러나, 이러한 지연은 LFSAR 제어기로부터의 요구 라인이 필요한 것보다 1 사이클 더 오랫동안 활성상태에 있음을 의미한다. RSC 우선순위 스테이션은 다음의 동작 동안 이들 요구를 분석할 시에 이를 고려함으로써, 동일한 동작을 다시 선택하는 사이클을 낭비하지 않을 것이다. 각 LFSAR 제어기(11)에 대해 두가지 유형의 승인 신호가 발생되는데, 이는 RSC 우선순위 스테이션에서 무슨 일이 발생하는가를 분명하게 나타낸다. 정규의 승인 신호는 LFSAR 요구의 선택시마다 발생되며, 특정 신속-경로는 파이프라인 신속-경로 명령이 선택되는 경우에 발생된다.

일단 RSC 우선순위 스테이션(21)이 인터페이스를 통해 전송하기 위한 명령을 선택하면, 그 명령 및 그의 연관된 정보(LFSAR ID, 요구기 ID 및 LFSAR 버퍼 ID)가 적절한 RSC 자원 레지스터에 로딩된다. 도 4는 적당한 자원 보유 레지스터내로의 명령 스테이징 방법을 도시한 것이다. CLC 명령(42) 및 C3 파이프 명령(43) 스테이징 레지스터의 목적은 도 4의 우측 상단 코너에 도시된 3-웨이 멀티플렉서(41)를 통하는 임계 경로의 타이밍을 완화시키고자 하는 것이다. 모든 CLC 명령은 LFSAR 제어기(11)로부터 나오며 타이밍은 임계적이다. 전체적인 시스템 성능의 향상을 위해, 원격 동작들이 단일 사이클로 인터페이스를 통해 전송된다. 이 동일한 명령은 우선순위 로직 및 작은 크로스포인트 스위치를 횡단하여 적당한 RSC 자원 레지스터에 이르러야 하므로, 챌린징 경로(challenging path)가 필요한데, 이는 양호한 실시예에서 입력 CLC 명령 및 파이프 명령을 스테이징한 후에 그들을 크로스포인트 스위치를 통해 전송하는 것에 의해 해결된다.

도 4를 더욱 상세히 살펴보면, 각각의 LFSAR 제어기(11) 요구와 연관된 명령들은 CLC 명령 스테이징 레지스터(42)내로 스테이징된다. 이와 병행하여, 파이프라인 사전 우선순위 스테이션(32)이 선택한 C2 파이프라인 명령은 C3 파이프 명령 스테이징 레지스터(43)내로 스테이징된다. 또한 이와 병행해서, CLC 파이프 명령은 3-웨이 멀티플렉서(41)를 통해 흐른다. 이 멀티플렉서는 명령 우선순위 스테이션(33)으로부터 나오는 승인 라인들에 의해서 제어된다. 각각의 CLC 명령은 그 CLC의 파이프라인과 연관된 2개의 RFAR 명령(45) 또는 RSAR 명령(47) 레지스터내로 로딩된다. 이는 각각의 CLC 명령이 4개의 가능한 목적지를 가짐을 의미한다. 신속-경로 명령은 CPU 인출 동작에 제한되므로, 그들 명령은 2개의 RFAR 명령(45) 레지스터내로만 로딩될 수 있다. 이들 명령은 2-웨이 멀티플렉서(44) 및 게이트웨이(gateway)(46)로 이루어진 크로스포인트 스위치를 통해 라우팅된다. 2-웨이 멀티플렉서는 CLC 명령(42) 레지스터 또는 C3 파이프 명령(43) 레지스터를 선

택하는 신호들에 의해서 제어된다. 게이트웨이(46)는 CLC 명령(42)을 통과시킬 수 있는 신호 게이팅 라인에 의해서 제어된다. 이들 제어 신호의 모두는 서로 직교하는 것으로서, 선택된 RSC 동작의 승인과 다음의 이용 가능한 자원을 선택하는 우선순위 로직과의 조합으로부터 발생된다.

RSC는 모든 크로스-클러스터 동작을 다루는 8개의 자원 레지스터를 포함한다. 따라서, 각각의 파이프라인에 대한 2개의 인출 동작 및 2개의 저장 동작이 동시에 발생될 수 있다. 전체적인 동작은 로컬 RSC 인터페이스 제어기(10)에 의해서 추적되므로, 동작의 완료에 필요한 모든 정보는 자원 레지스터(24)내에 보유되어야 한다. 도 5는 디스에이블 비트, 유효 비트, 명령 레지스터, 오리지널 요구기 ID 레지스터, LFSAR 제어기 ID 레지스터 및 LFSAR 버퍼 레지스터로 구성된 단일 세트의 자원 레지스터(59)를 상세히 도시한 것이다. 양호한 실시예에서는, LFSAR 제어기(11)내의 각 파이프에 대해 2개의 LFAR 및 2개의 LSAR이 존재하며, 각 파이프에 대해 2개의 LFSAR 버퍼만이 존재한다. 그러므로, LFSAR 제어기(11)는 새로운 동작마다 각각의 LFSAR ID에 두 버퍼 중의 하나를 동적으로 할당해야 한다. 따라서, 버퍼 ID는 각각의 새로운 요구와 함께 RSC IC(10)에 전송되어야 한다.

주목해야 할 것은 당업자라면 LFSAR 자원 및 버퍼의 총 수와 그들의 상호 관계가 어떤 식으로 본 발명에 영향을 끼치지 않는 지를 이해할 수 있다는 것이다. 버퍼의 수가 LFSAR 자원의 수와 동일하고 그들간의 관계가 일정한 경우, RSC IC는 그 정보를 추적하기 위해 별도의 자원을 필요로 하지 않을 것이다. 그러나, RSC IC가 양호한 실시예에서 설명하는 정보 이상의 부가적인 정보를 추적할 필요가 있는 다른 실시예들이 있을 수도 있다. 이들 다른 실시예에는, 자원 레지스터에서 추적되어야 하는 정보의 양에 관계없이, 여기에 개시한 원리들이 여전히 적용될 수 있다.

도 5를 다시 참조하면, C3 신속-경로 스테이징 레지스터(55a) 및 CLC 스테이징 레지스터(55b)의 모두를 구비하고 있기 때문에 약간 더 복잡한 RFAR 자원이 상세히 도시된다. 도 4에는 자원 레지스터의 명령 부분을 도시하되 8개의 모든 자원 레지스터를 도시했다. 반면에, 도 5에는 단일의 자원을 도시하되 주어진 RSC 동작의 추적에 필요한 모든 정보를 자원 레지스터내로 로딩하는 방법을 도시한다. 모든 제어 신호는 명령 우선순위 스테이션(33)으로부터 나온다. 도 5에서는 (사용되는 실제 신호의 서브세트인) 우선순위 승인만을 이용해서 각종 자원 레지스터의 로딩 방법에 대한 로직 타이밍을 설명한다.

먼저, 유효 비트(59a) 동작에 대한 승인이 발생된 후의 사이클에서 로딩된다. OR 게이트(54)는 동작이 정규의 CLC 동작인지 또는 파이프라인 신속-경로 동작인지의 여부에 관계없이 유효 비트가 로딩되게 한다. 유효 비트들은 자원이 이용가능한지의 여부를 판단하는데 있어 중요한 역할을 하므로, 그 유효 비트는 자원이 다음의 우선순위 사이클에서 이용가능하지 않은 것으로 표시되게 한다. 승인을 자원 레지스터(59)의 전체 세트에 제공하는 것으로 인해서 유발되는 타이밍 임계 경로를 완화시키기 위해, 본 발명에서는 잔여 정보를 로딩전에 지연시킬 수 있다는 사실을 이용한다.

LFSAR 제어기(11)로부터 나오는 CLC 동작은 로직 타이밍면에서 가장 간단하다. 이들 동작에 있어, 스테이징된 CLC 승인(53)은 2-웨이 멀티플렉서(56) 및 게이트(58)를 OR 게이트(54)를 통해 제어한다. 따라서, 그 승인이 활성상태인 경우, CLC 스테이징 레지스터(55b)를 포함한 모든 정보는 유효 비트(59a)의 로딩 후의 사이클에서 잔여 자원 레지스터 세트(59b-59d)내로 로딩된다.

파이프라인 신속-경로 동작은 C3 명령 및 요구기 ID 레지스터만을 갖는 C3 신속-경로 스테이징 레지스터(55a)내로 스테이징된다. 이러한 상황에서, 스테이징된 신속-경로 승인 LC(FP LC)(52a)는 2-웨이 멀티플렉서(56)를 통해 C3 파이프라인 명령 및 요구기 ID를 선택하여 그들을 명령(59b) 및 요구기 ID(59c) 자원 레지스터내로 로딩한다. 일단 이들 신속-경로 동작들이 파이프라인의 제 3 스테이지에 도달하면, 그들 동작은 LFAR 자원내로 로딩되고 LFAR 버퍼에 할당된다. 이것이 필요한 이유는 대다수의 CPU 인출 동작이 다수의 파이프라인 경로를 필요로 하여 LFAR이 전체적인 인출 동작의 관리에 필요하기 때문이다. 그러므로, 일단 이러한 할당이 알려지면, LFSAR 제어기(11)는 파이프라인 신속-경로 동작을 바로 뒤따르는 사이클에서 RSC IC(10)에 특정 요구를 발생할 것이다. 이 요구와 함께, LFSAR 인터페이스는 LFAR ID 및 LFAR 버퍼 ID를 포함할 것이다. RSC IC는 신속-경로 승인 LC2 트리거를 이용하여 정보가 CLC BFR 및 CLC REQ 레지스터(55b)에서 이용가능할 때를 정한다. 따라서, 그 정보는 명령(59b) 및 요구기 ID(59c) 레지스터를 뒤따르는 사이클에서 게이트웨이(58)를 통해 게이팅되어 LFSAR 버퍼(59d) 및 LFSAR ID(59e) 레지스터내로 로딩될 수 있다.

양호한 실시예에서는 또한 CLC REQ 레지스터내에 상주하는 8개 요구 신호의 8/3 인코딩을 수행하여 ID를 LFSAR ID 레지스터(59e)내에 3 비트 값으로서 저장하는 인코더(57)의 사용에 대해 개시하고 있다. 디스에이블 레지스터(59f)로서 표명되는 하나의 부가적인 비트도 완전성을 위해 포함된다. 이 단일 비트 레지스터는 양호한 실시예의 공용 버스(UBUS)를 통해 또한 로딩될 수 있는 스캐너블 레지스터(scannable register)이다. 각각의 RSC 자원 레지스터는 자원이 마이크로코드, 펌웨어 로드, 시스템 리세트 등을 통해 영구적으로 또는 임시적으로 디스에이블될 수 있게 하는 디스에이블 비트를 가진다. 디스에이블 비트가 RSC IC(10)의 정상적인 시스템 동작에서 어떠한 역할을 하지는 못하나, 디버그 및 인터페이스 성능 분석을 설계하는데 있어서의 보조자로서 역할한다.

도 1b에 도시된 응답 처리기(22)는 원격 클러스터로부터 복귀하는 모든 응답 통신 신호를 처리하여 개시 프로그램에 적정한 완료 신호를 전송하는 임무를 담당한다. RSC IC(10)가 수신하는 주 형태의 응답은 두가지가 있다. RSC 동작의 대부분은 원격 교차 질의(XI)를 포함하여 데이터가 원격 캐쉬내에 상주하는 지를 판단한다. 이들 동작중의 하나가 호출될 때마다, 명령은 다른 사이드에서 수신되어 보장된 우선순위 레벨에 의해서 원격 파이프내로 들어간다. 파이프내로 들어가는 그 보장된 엔트리는 인터페이스를 통해 동작이 발생되는 시점과 적중/적중실패가 알려진 시점간에 동기 관계가 존재할 수 있게 한다. 양호한 실시예에서, XI 응답은 RSC 인터페이스에 명령이 제시된 후의 4개 사이클에서 복귀된다.

RSC 우선순위(21) 사이클에서 시작하여, 응답 처리기(22)는 도 6에 도시된 스테이징 메카니즘에 의해 XI 응답을 추적한다. 로컬 캐쉬에 대한 적중실패를 발생하는 어떤 유형들의 인출에 대한 성능을 향상시키기 위해, RSC IC는 인출이 동기 인터페이스에 결합된 로컬 L3 또는 원격 L3 메모리를 타겟으로 하고 있는지의 여부를 나타내는 어드레스 비트를 사용하여 그 동작을 자동적으로 퇴진시킬지의 여부를 판단한다. 예를 들어, 인출이

원격 L3 메모리를 타겟으로 하고 있는 경우에는 동작이 완료될 때까지 RSC 자원이 계속 유효 상태로 유지되는데 이는 원하는 데이터가 원격 캐쉬 또는 원격 L3로부터 나올 것이기 때문이다. 그러나, 어드레스가 로컬 L3을 타겟으로 하고 있고 데이터가 원격 캐쉬내에 상주하지 않는 경우에는 그 자원이 새로운 동작의 작업에 대해 자유롭게 될 수 있는데 이는 데이터 인출이 로컬 LFAR 제어기에 의해서 처리될 수 있기 때문이다.

매 사이클에서 각각의 CLC 명령 레지스터(55b)를 디코더(61)에 의해 분석하여 그 명령이 교차 질의(XI)를 필요로 하는 인출 명령들 중의 하나 인지의 여부를 알아 본다. 그 결과는 CLC 로컬 L3 비트(60) 및 CLC 동작에 승인이 발생되었음을 나타내는 RSC 우선순위 스테이션(21)으로부터의 신호와 조합된다. 이와 병행하여, C3 파이프 로컬 L3 비트는 신속-경로 동작에 승인이 발생되었음을 나타내는 RSC 우선순위 스테이션(21)으로부터의 신호와 비교된다. 자명한 일로서, 모든 신속-경로 동작에는 교차 질의가 필요하다. 승인들은 상호 배타적이므로, 어떤 사이클에서 하나의 브랜치만이 활성화될 수 있다. 이들 신호는 2-웨이 AND/OR 멀티플렉서(63)내에서 도시된 바와 같은 식으로 조합되며, 조건들이 참(true)인 경우에 4-비트 L3 스테이징 파이프라인(64)의 1 비트가 로딩된다. 이 파이프라인은 스테이지 2로 시작하여 스테이지 6으로 끝나는 각각의 사이클에 대해 4-비트 스테이징 레지스터를 포함한다. 4개 비트의 각각은 RFAR 자원(12a) 중의 하나를 나타낸다. 도 6에 도시하지 는 않았으나, 방금 설명한 기능을 포함하는 모든 요소들은 4회 복제되며 결과의 출력은 스테이지 2의 각 비트에 공급된다. RFAR 자원들 중의 하나만이 어떤 주어진 사이클에서 로딩될 수 있으므로, L3 스테이징 파이프라인(64)내의 각 스테이지의 4개 비트들은 서로 직교한다. 파이프라인의 스테이지 6은 XI 응답이 응답 처리기(22)에 의해서 수신되는 사이클에 대응한다. 4개 비트중의 어떤 비트가 활성 상태이고 XI 응답이 적중실패에 해당하는 경우, 대응하는 RSC 자원 레지스터는 유효 비트(59a)의 턴오프에 의해서 리세트된다.

데이터 인출 서브세트 동안에만 로딩되는 특정의 L3 스테이징 파이프라인(64)외에도, 도 6은 매번 새로이 시작되는 RSC 동작과 함께 로딩되는 RFSAR 스테이징 파이프라인(67)을 도시한다. 각각의 RSC 자원 레지스터(24)는 최종 사이클에서 자원이 로딩되는 것을 나타내는 단일 비트 래치를 구비한다. 8개의 자원 로드 래치(65)는 하나의 자원만이 각각의 사이클에서 새로운 동작과 함께 로딩될 수 있으므로 서로 직교한다. 그들 8개 레지스터의 출력은 8/3 인코더(66)에 의해 인코딩되며, 3-비트의 인코딩된 값은 RFSAR 스테이징 파이프(67)내에 저장된다. 이 파이프라인도 스테이지 2로 시작해서 스테이지 6으로 끝난다. 유효 비트와 결합된 3-비트 RSC 자원 ID는 스테이지 6에 도달할 때까지 각각의 사이클에서 스테이지들을 통해 전송된다. 로직 타이밍은 XI 응답 및 원격 인터페이스 에러 신호가 수신되는 사이클인 스테이지 6에 RSC 자원 ID이 도달되게 하는 타이밍이다.

인터페이스 에러의 경우, 3-비트 RFSAR ID는 디코딩되어 동작에 연루된 RSC 자원 레지스터(21)의 리세트에 사용된다. 또한, 하드웨어 룩업 테이블(68)은 3-비트 RSC ID를 사용하여 그 자원 레지스터의 LFSAR ID(59e) 레지스터를 인덱싱하는데 이용된다. LFSAR ID 레지스터의 내용들은 또한 디코딩되어 적정한 LFAR 또는 LSAR 제어기에 인터페이스 에러 신호를 전송하는 사용된다. 예를 들어, RFSAR 스테이징 파이프라인(67)의 스테이지 6이 "010"의 값을 포함하는 경우, 이는 파이프 A의 RSAR 00이 RSC 자원임을 나타낸다. 룩업 테이블(68)은 그다음 파이프 A RSAR 0 자원내의 LFSAR ID 레지스터를 디코딩할 것이고 그 값은 그 특정 동작과 연관된 LSAR을 가리킨다. 주어진 동작을 대응하는 로컬 LSAR 또는 LFAR 제어기와 연관시키는 능력 덕분에, 대부분의 RSC 동작들이 재시도될 수 있다. 인터페이스 에러는 간헐적이므로 그 동작의 재시도를 위한 능력은 불필요한 시스템 중단을 방지한다.

RFSAR 스테이징 파이프라인(67)을 이용하는 하나의 부가적인 시스템 성능 향상으로서는 신속한 판독 전용 무효화가 있다. RSC 동작들 중의 하나는 캐쉬내에 상주하는 판독 전용 데이터 카피를 원격 사이드에서 무효화하는 판독 전용 무효화이다. 이러한 무효화는 예를 들어 CPU가 독점권을 가진 데이터를 인출하고자 하고 다른 CPU들이 판독 전용 카피를 갖는 경우에 발생한다. 다른 CPU들이 원격 사이드에 있으면, RSC IC는 다른 클러스터의 원격 인출 제어기(12a)가 처리하는 판독 전용 무효화 명령을 전송할 것이다. 통상적으로, 이것에 의해서, 디렉토리 엔트리를 무효화하기 위한 파이프라인 통과가 간단하게 된다. 이따금, 그들 초기 파이프라인 통과들에 의해서, 원격 CPU대신에 동일 라인을 액세스하려고 하는 다른 제어기에 대한 어드레스 비교가 발생한다. 이들 충돌이 발생하면, 그러한 충돌이 완전하게 해결되기 전에 독점권을 요구하는 CPU가 데이터를 가질 수 있게 하는 것이 안전한 시기들이 있다. 본 발명의 원격 인출 제어기(12a)는 초기 파이프라인 통과 동안 그들 "안전한" 시나리오를 검출하여 처리가 안전한 동기 XI 응답 버스를 통해 RSC IC에 통보한다.

도 7은 다른 두 개의 상술한 스테이징 파이프라인과 유사한 방식으로 작동하는 판독 전용 무효화 스테이징 파이프라인(75)을 도시한 것이다. 두 파이프라인에 대한 CLC 명령 레지스터(55b)는 판독 전용 무효화 동작들을 필터링하는 ROI 디코더(73)에 의해 디코딩된다. 이것은 CLC 승인(53)과 결합되어 유효 판독 전용 동작이 개시되었음을 나타낸다. 어떤 주어진 사이클에서 이들 동작중의 하나만이 개시될 수 있다. 그 결과는 ROI 멀티플렉서(74)에 공급되어 판독 전용 스테이징 파이프라인(75)의 스테이지 2를 세팅하는데 사용된다. 이 비트는 스테이지 6으로 전송되어, 여기서 응답 처리(22)에 수신된 XI 응답과 정렬된다. RO 무효화 스테이지 6 비트가 활성 상태이고 XI 응답이 적중실패인 경우, RFSAR 스테이지 6 레지스터(63) 및 LFSAR 룩업 테이블(68)은 관련 LFAR 제어기를 해제시켜 초기 동작을 완료할 수 있게 하는데 이용된다. 원격 인출 제어기는 판독 전용 무효화를 계속 처리하여 RSC자원 유효 비트가 활성 상태로 유지되게 한다. 일단 원격 인출 제어기(12a)가 동작을 완료하면, 그 제어기는 그 동작을 재시도하여 자원이 새로운 작업을 받아들일 수 있게 하는 최종 응답을 복귀시킨다. 그 사이에, 판독 전용 무효화와 연관된 LFAR은 새로운 동작을 시작했을 수도 있다. 판독 전용 무효화에 대한 최종 응답이 새로운 LFAR 동작에 대한 최종 응답으로 오인되지 않도록, RSC IC는 각각의 RFAR 자원에 대한 보유 레지스터를 구비한다. 적당한 보유 레지스터는 신속 판독 전용 무효화 메카니즘이 LFAR의 해제에 사용될 때마다 세팅되어, 곧 다가오는 최종 응답이 LFAR에 전송되지 못하게 한다. 일단 최종 응답이 수신되고 동작이 공식적으로 완료되면, 보유 레지스터는 잔여 자원 레지스터와 함께 리세트된다.

교차 질의는 아니고 원격 동작을 포함하는 본 발명의 모든 동작은 인코딩된 최종 응답으로 종료된다. 응답 처리기(22)는 인코딩된 응답 ID 버스를 사용하여 그 응답을 동작을 개시한 LFSAR ID와 일치시킨다. 최소한, RSC IC(10)는 동작 완료를 발신 LFSAR 제어기에 알려 그 제어기가 그의 자원을 해제할 수 있게 한다. 데이터가 원격 사이드로부터 인출되는 경우, 데이터 어드밴스(data advance)가 대응하는 로컬 LFAR 제어기에 전송

됨으로써 그 제어기는 로컬 디렉토리 상태를 갱신할 수 있다. 또한, RSC XPT 코드포인트가 데이터플로우 칩(dataflow chip)으로 전송될 수 있게 하는 신호들이 XPT 제어기(25)로 전송된다.

원격 동작들의 서브세트는 또한 전체 응답 코드가 CFAR 제어기(15)로 전송하는 것을 필요로 한다. 예를 들어, CFAR 제어기(15)는 그 신호 응답을 사용하여 초기 및 최종 응답을 중앙 처리 유니트에 전송해야 한다. 본 발명에서 규정한 7개 응답 비트들 중의 비트 0 및 비트 1은 실제 응답 값에 포함되지 않는다. 대신에 그들 비트는 다음의 특정 의미를 갖는다.

* 비트 0은 원격 동작들이 거절되었음을 나타내어 통상적으로 데드록 상황을 방지한다. 이 비트는 재시도 신호가 적정한 LFSAR에 전송되게 한다. LFSAR는 차후 동작에 대한 재시도를 시도할 것이다.

* 비트 1은 라인의 원격 캐쉬에 대한 적중이 변경된 상태로 발생한 것을 나타낸다. 이 정보는 로컬 디렉토리의 최종 상태의 계산을 위해 데이터 인출 동안 LFAR에 의해서 사용된다.

잔여 비트들은 초기 동작에 따라 각종 완료 코드를 나타내기 위해 인코딩된다.

원격 사이드로부터 복귀하는 응답들의 처리외에도, RSC IC는 또한 응답 우선순위 기능을 이용하여 응답을 원격 사이드로 전송한다. 이들 응답은 교차 질의(XI) 및 원격 클러스터로부터 개시되어 로컬 클러스터에서 처리되는 동작들에 대한 최종 응답들의 형태를 가진다. (총 4개 요구기에 대한) 각 파이프라인으로부터의 로컬 RFAR(12a) 및 RSAR(12b) 제어기는 XI 응답 및 요구를 제시하여 최종 응답을 RSC IC(10)로 전송한다. 교차 질의는 인출 동작들에만 관계되므로, RFAR 제어기(12a)만이 XI 응답을 제시할 수 있다. 또한, 한 번에 하나의 교차 질의만이 원격 사이드에 의해 개시될 수 있고 교차 질의는 일정 수의 사이클에서 파이프라인을 통해 처리될 수 있으므로, 4개의 가능한 RFAR XI 응답중에서 하나만이 어떤 주어진 사이클에서 활성 상태일 수 있다. 따라서, 응답 우선순위(23) 로직은 간단히 4개의 응답을 OR 연산한 그 출력을 인터페이스로 전송한다.

최종 응답 요구들은 RFAR(12a) 및 RSAR(12b) 제어기로부터 나올 수 있으며, 원격 동작의 길이가 다양하므로 응답은 비동기적으로 발생한다. 응답 우선순위(23) 로직은 RSC 우선순위 스테이션(21)과 상호작용하여 최종 응답들 중의 어떤 것이 우대될 수 있는지의 여부를 판단한다. 데이터 인출이 아닌 다른 동작의 경우, 그 응답 로직은 간단한 우선순위 동작을 사용하여 4개 RFSAR 중의 하나를 선택하여 그 응답을 인터페이스를 통해 전송한다. 둘 이상의 RFSAR이 동일 사이클에서 요구를 발생하면, 그 동작은 RSAR보다는 RFAR에 대해 행해진다. 따라서, CPU가 필요로 하는 인출 데이터가 RSAR 응답 통신에 의해 불필요하게 지연되지 않게 함으로써 시스템 성능이 향상된다. 두 RFAR이 동일한 사이클에서 요구를 제시하는 경우에는, 그 동작은 라운드 로빈 방법을 사용하여 그들 RFAR 중의 하나를 선택한다. 어떠한 RFAR도 우대될 수 없고 둘 이상의 RSAR이 요구하고 있는 경우에는, 간단한 라운드 로빈 방법에 의해 RSAR이 선택된다.

본 발명의 새로운 관점들 중의 하나는 응답 우선순위 기능(23)과 RSC 우선순위 스테이션(21)이 상호작용하여 공유 데이터 버스의 효율을 최대화하는 것이다. (원격 개시 인출에 관련하는) 복귀 인출 데이터는 로컬 개시 저장 동작과 동일한 데이터 경로를 공유해야 하므로, 인출이 지연되는 한편 저장 전달의 완료를 기다려야 할 가능성이 존재한다. 응답 우선순위는 이러한 가능성을 인출 데이터의 복귀를 시도하는 최종 응답 요구에 대한 다음의 단계들을 수행하는 것에 의해서 감소시킨다.

1. 우선순위 로직은 요구하는 RFAR의 파이프라인에 대응하는 데이터 경로가 이용가능한지의 여부를 알아보기 위한 검사를 행한다. 이때, 이용가능하지 않은 경우, 우선순위 로직은 RSAR 요구가 사용중에 있지 않으면 그 요구를 즉시 선택할 것이다.

2. 데이터 경로가 이용가능한 경우에는, 우선순위 로직은 RFAR을 선택하고 RSC 우선순위 스테이션에 통보하여 데이터 전달을 포함하는 어떤 LSAR 저장 동작의 선택도 차단한다.

두 우선순위 기능(21, 23)에서의 동작은 순환적이고 동적인데, 이것은 그들이 각 사이클에서 현재 환경을 평가하여 단일 사이클에서 모든 결정을 함을 의미한다. 그러므로, 요구가 데이터 경로의 이용불가능성으로 인해 지연되는 경우, 그 요구는 서비스 가능한 제 1 사이클에서 서비스될 수 있다. 인터페이스를 통한 전달을 위한 요구가 선택될 때마다, 그 요구하고 있는 RFSAR에 승인이 전송됨으로써 그 요구하고 있는 RFSAR은 현재 요구를 배제시키고 새로운 요구를 다음 사이클에서 발생할 수 있다. 실제 응답외에도, RSC IC는 또한 RFSAR이 응답을 복귀시키는 것을 나타내는 3-비트 인코딩된 응답 ID 버스를 전송한다. 다른 사이드의 응답 처리기(22)는 이 3-비트 ID를 디코딩하여 어떤 RSC 자원 레지스터의 리세트 필요성을 해결한다.

모든 응답은 단일 사이클에서 RSC 인터페이스를 횡단하나, 두 동작은 부가적인 정보를 위해 응답 버스를 이용한다. 키 동작 동안, 실제 키는 응답을 즉시 뒤따른다. 테스트 바이트 절대(Test Byte Absolute: TBA) 동작 동안, TBA 상태는 다음 사이클에서 응답을 뒤따른다. 어떤 경우에서도, RFSAR 제어기는 특정 신호를 그것이 2-사이클 동작임을 나타내는 응답 요구를 수반하는 RSC IC에 전송한다. 따라서, 응답 우선순위(23)는 어떤 새로운 RFSAR 최종 응답이 제 2 사이클 동안 선택되지 않게 할 수 있다.

양호한 실시예의 RSC 인터페이스는 총 4개의 쿼드워드(QW) 데이터 경로를 지원한다. 각각의 파이프에 대해서는 각 방향으로(로컬로부터 원격으로 또는 원격으로부터 로컬 방향으로) 하나의 데이터 경로가 있다. 물리적으로 각각의 데이터 경로는 그의 구현을 위해 2개의 SCD 칩(IBM의 저장 제어기 데이터 플로우 칩)을 필요로 한다. 이들 데이터 칩의 각각은 더블워드(DW) 데이터를 보유한다. 이 구성은 공유 버스 구조와 전용 포인트-포인트 데이터 흐름을 절충한 구성이다. 각 방향으로 단방향 데이터 경로가 있지만, 각각의 데이터 경로는 인터페이스의 양쪽 사이드로부터 개시되는 데이터를 다중화해야 한다. 예를 들어, 원격 SC를 로컬 SC에 연결하는 데이터 경로는 로컬 개시 인출 요구에 응답해서 데이터를 복귀시키기 위해 어떤 시점에서 사용될 수도 있고, 또는 원격 사이드에 의해 개시되는 저장 동작을 수반하는 저장 데이터를 전달하는데 사용될 수도 있다. 이들 동작은 개별 데이터 경로에 의해 분리되는 것이 이상적이나, 패키징 제약 요인으로 인해서 그렇게 할 수 없다. 그러나, 단방향 버스들이 각 파이프에 대해 양방향으로 존재하므로, 사이클마다 4개의 QW(64 바이트)가 동시에 이동될 수 있다.

RSC IC는 4개의 모든 데이터 경로를 관리하는 크로스포인트(XPT) 제어기(25)를 구비한다. 실제, 각 데이터

경로의 절반은 각 클러스터의 XPT 제어기에 의해서 제어된다. 예를 들어, 로컬 SC로부터 원격 SC로 전달되는 데이터는 로컬 RSC IC에 의해서 구동되며 원격 RSC IC에 의해서 수신된다. 따라서, XPT 버스의 구동 부분은 로컬 RSC IC로부터 나오며 수신 부분은 RSC IC로부터 나온다. 모든 4개의 데이터 경로는 11-비트 제어 버스에 의해서 제어된다. 이들 비트중의 비트(0:5)는 수신 사이드를 제어하며, 비트(6:10)는 구동 사이드를 제어한다. 이들 부분적인 XPT 버스는 이후 수신 XPT(RCV XPT) 및 구동 XPT(DRV XPT)라고 한다.

도 8은 하나의 파이프라인이 하나의 RSC IC내에서 XPT를 수신하고 구동하는 내부 로직을 도시한 것이다. 임계 타이밍 경로의 완화를 위해, XPT 정보는 가능한 때마다 미리 셋업된다. XPT GEN(81a) 로직의 임무는 RSC 자원 정보 및 외부 신호를 사용해서 적당한 데이터 경로 제어를 셋업하고자 하는 것이다. 트리거(82a, 82b)는 가동 데이터에 대해 정확한 시점에서 XPT 정보를 RSC XPT 버스상으로 해제하는 게이트로서 작용한다. RSC XPT의 비트들은 데이터 칩의 각종 버퍼 제어기 및 크로스포인트 스위치에 의해서 수신된다. 이 로직은 버퍼 어드레스 및 기록 제어와 선택기를 작동시키는 간단한 디코더들로서 이루어진다. 데이터 칩들은 데이터 전달 후의 논리적 동작을 알지 못하므로, RSC XPT 버스는 각각의 QW가 전달되고 있는 동안 일 회 "펄싱된다(pulsed)". 따라서, 데이터 라인이 이동할 필요가 있으면, rsc ic는 16개의 연속 사이클 동안 RSC IC 버스상에 적당한 값을 유지해야 한다.

XPT 제어기(25)의 RCV_XPT 부분을 참조하면, RCV XPT GEN(81a) 로직에 RFAR 자원 레지스터, 차단_xpt_신호 및 RSC_CMD가 공급된다. 상술한 바와 같이, 데이터는 로컬 개시 요구로부터의 복귀 인출 데이터 또는 원격 개시 저장 동작으로부터의 입력 저장 데이터와 같은 두가지 이유에서 SC에 의해서 수신될 수 있다. 전자의 경우, RSC IC는 인출 동작을 담당하며 한 세트의 RFAR 자원 레지스터내의 모든 정보를 갖는다. XPT GEN 로직은 이 정보를 Cmd(59b), Req ID(59c), LFAR 버퍼(59d) 및 LFAR ID(59e) 레지스터에서 사용하여 RCV_XPT의 값을 계산하고 데이터 전달 길이를 결정한다. 데이터 전달 길이가 1 QW보다 큰 경우에는 XPT_CNTR(83)이 적정 수의 사이클에 의해 로딩된다. 이러한 로딩은 명령이 인터페이스를 통해 디스패칭된 후 짧게 발생한다. 데이터의 복귀시, 제 1 QW는 언제나 '03'x, '23'x, '05'x 또는 '18'-1B'x의 인코딩된 응답을 수반한다. 이들 응답 중의 하나가 (정합 enc_resp_id와 함께) 트리거 로직(82a)내로 수신되면, RCV_XPT의 RSC_XPT 버스상으로의 해제가 트리거된다. 다수의 QW가 관계되는 경우, XPT_CNTR은 카운트가 소멸될 때까지 RCV_XPT 값을 계속 공급할 것이다. RCV_XPT는 동작에 따라 (이후 캐쉬내에 포함시키기 위해) 데이터를 적정한 CPU 포트, I/O 포트, LSAR 버퍼 및/또는 LFAR 버퍼로 배향시킬 것이다. 어떤 상황에서는, 로컬 CPU을 목적지로 하는 데이터가 최종 순간에 CPU로 전송되지 않는다. RSC IC는 트리거의 억제를 위해 사용되는 각각의 LFAR(11) 및 CFAR(15) 제어기로부터 수개의 차단 신호를 수신한다.

RCV_XPT 버스의 사용에 관한 제 2의 시나리오에는 원격 클러스터로부터 개시된 저장 데이터를 수신하는 것이 포함된다. 이는 완전히 비동기적인 이벤트이므로, XPT_GEN(81a) 및 트리거 로직(82a)이 동시에 호출된다. RSC IC는 로컬 RSAR 제어기(12b)로부터의 데이터 어드밴스 트리거 및 입력 RSC_CMD 버스의 일부를 감시한다. RSAR_DADV가 활성 상태이고 명령의 서브세트가 적당한 값으로 디코딩되면, RCV_XPT는 셋업되고 RSC XPT에 제시되어 로컬 데이터 경로 칩들이 입력 데이터를 받아들여 그 데이트를 RSAR 버퍼로 라우팅하도록 할 수 있다.

구동 XPT(DRV_XPT)는 유사한 방식으로 작동한다. 이러한 데이터 경로의 사용은 두가지 시나리오에서 필요하다. 한가지 시나리오는 로컬 개시 저장 동작에 관련된다. 이 경우, RSAR 자원 레지스터들은 필요한 때 DRV_XPT를 셋업하고 XPT_CNTR(83)를 로딩하는데 필요한 정보를 보유한다. 트리거 로직(82b)이 수신하는 RSAR_ST_OP 신호는 인터페이스를 통한 명령의 발생에 대해 데이터를 이동시키기 시작해야 하는 시기의 타이밍을 제어한다. 다른 사이드의 RSAR는 RSAR_DADV를 원격 RSC IC에 대해 작동시켜 그 RSC IC가 "기상(wake up)"하여 저장 데이터를 받아들일 수 있게 한다. RSAR_ST_OP 신호는 현재 동작이 저장 데이터 전달을 필요로 하는 지를 판단하기 위한 RSAR 명령(59b) 레지스터의 간단한 디코드이다. 모든 데이터는 로컬 사이드의 LSAR 버퍼들로부터 나오며, DRV_XPT는 이들 버퍼의 판독을 제어한다.

다른 시나리오는 다른 사이드가 요구하는 인출 데이터를 복귀시키는 것에 관련된다. 이 데이터의 공급원은 RFAR 버퍼, 주 메모리 어댑터(PMA) 인터페이스 또는 CPU 원격 센스 레지스터(Remote Sense Register)일 수 있으므로, XPT GEN(81b) 로직은 MBA_ID외에도 RFAR 제어기(12a)로부터 나오는 신호들의 조합을 사용한다. 비-제로(non-zero) MBA_ID는 데이터가 ID에 대응하는 원격 센스 레지스터로부터 나오는 것을 의미한다. ID가 제로이면, 각종 RFAR 신호들은 데이터가 RFAR 버퍼 또는 PMA 인터페이스로부터 나오는 지의 여부를 판단하는데 사용된다. 이들 신호 중의 하나인 PMA_DATA_RDY 신호는 데이터가 L3 메모리로부터 저장 제어기로 전달되고 있는 기간동안 RFAR 제어기에 의해서 발생된다. 응답 우선순위(23)가 그 기간 동안 RFAR 인코딩된 응답 요구를 처리할 수 있으면, 데이터는 RFAR 버퍼를 우회하고 직접적으로 RSC 인터페이스로 전달될 수 있다. 한편, 그 기간의 끝에 이른 후에 응답 우선순위(23)가 요구하고 있는 RFAR에 승인을 발생하면, PMA_DATA_RDY 신호는 배제된다. 이때, XPT GEN(81b) 로직은 RSC 인터페이스가 이용가능하게 되어 버퍼로부터의 데이터가 그 인터페이스로 이동될 수 있을 때까지 데이터를 버퍼로 라우팅할 것이다. 본 발명의 이러한 관점은 원격 클러스터에 대한 CP 인출 동안 불필요한 버퍼 로딩 및 언로딩을 배제시켜 시스템 성능을 더욱 향상시킨다.

DRV_XPT의 셋업외에도, RFAR 제어기(12a)로부터의 수개의 신호들은 데이터 전달 길이가 도출될 수 있게 하는 CRF_XFR_LEN 버스를 구비한다. 복귀 데이터의 경우, 트리거 로직(82b)은 "복귀 인출 데이터"를 나타내는 ENC_RESP 값과 결합된 응답 우선순위 스테이션으로부터의 RFAR 승인에 의해서 작동된다. 따라서, DRV_XPT는 RSC_XPT 버스의 제 2의 절반상으로 해제될 수 있다. 데이터 전달 길이가 하나의 QW보다 크면, XPT_CNTR(83)은 카운트가 소멸될 때까지 RSC_XPT를 계속해서 작동시킨다.

주목해야 할 것은 RSC의 비동기 특성으로 인해 로컬 RSC가 저장 동작을 개시하려고 함과 동시에 원격 인출 동작을 위한 데이터를 복귀시키려고하는 바와 같은 충돌이 빈번하게 일어 난다는 것이다. 데이터 경로상의 충돌을 피하고 성능을 극대화시키기 위해, XPT 제어기(25)는 우선순위 스테이션과 상당히 긴밀하게 상호작용하여 복귀 인출 데이터가 가능한 우선순위를 갖도록 한다. 또한, 일단 데이터 경로가 사용중에 있으면, 우선순위 스테이션은 클러스터들간에서 작업이 항상 이동하도록 하기 위한 데이터 경로를 필요로 하지 않는 새로운 동작들의 개시에 즉시 집중한다.

데이터 경로들은 목적지 및 공급원에 대한 제각기의 코드 포인트 정의를 가진 구동기(예를 들어 RCV_XPT 및 DRV_XPT)라고 불리우는 RSC IC(10)에 의해서 관리된다.

RSC 자원 레지스터의 각각은 도 5에 도시한 단일 비트 디스에이블(59f) 래치를 구비한다. 이 래치는 자원들의 어떠한 조합도 영구적으로 디스에이블되도록 '1'로 스캐닝될 수 있다. 또한, 이들 래치는 UBUS 레지스터에서 4개의 비트를 사용하는 것에 의해 또한 세팅될 수 있다. 양호한 실시예의 저장 제어기는 펌웨어 및 CP 밀리코드에 의해 판독, 기록 및 변형될 수 있는 일련의 UBUS 레지스터를 구비한다. 디스에이블 래치는 그들 밀리코드 제어가능 UBUS 레지스터를 통해 제어될 수 있으므로, RSC 자원들의 동적 디스에이블링은 밀리코드 루틴 또는 임시 패치의 일부로서 달성될 수 있다. 그의 한가지 용도는 각종 워크로드(workload)에 대한 복제 자원의 효과를 판단하기 위한 성능 비교 분석일 것이다. 코드 포인트는 RSC IC에서 그들 디스에이블 스위치에 대해 발생하는 결과적인 조치들을 제어한다.

낮은 코드 포인트('1'x-'6'x)는 '8'x-'F'x의 코드 포인트들과 다르게 작동한다. '8'x-'F'x 코드포인트가 호출되면, 선택된 자원은 RSC IC내에서의 관련 디스에이블 비트의 활성화에 의해 디스에이블링된다. 연속적인 UBUS 기록 동작들은 어떤 원하는 조합으로 다수의 자원들을 디스에이블링하는데 사용될 수 있다. '1'x-'6'x의 낮은 코드 포인트는 RSC IC내의 우선순위 로직이 디스에이블 모드를 감시하여 적절한 방식으로 인터페이스 활동을 제한하도록 하는 디스에이블 시나리오를 발생한다. 예를 들어, 모드 '2'x가 선택되면, 우선순위 로직은 제 1 동작이 완료될 때까지 제 2 동작이 발생되지 않도록 한다.

이제까지 본 발명의 양호한 실시예들을 설명하였으나, 당업자라면 다음의 특허청구범위의 범주내에 속하는 각종 변형 및 변경 실시예가 가능함을 알 수 있을 것이다. 따라서, 다음의 특허청구범위는 상술한 발명을 적절히 보호하고자 하는 취지로 해석되어야 할 것이다.

발명의 효과

이상 설명한 바와 같이, 본 발명에 따른 대칭적 멀티프로세싱 환경에서 자원들을 관리하는 원격 자원 관리 시스템은 전술한 종래 기술의 문제점들을 해결한다.

(57) 청구의 범위

청구항 1. 대칭적 멀티프로세싱 환경(symmetrical multiprocessing environment)에서 자원(resources)들을 관리하는 원격 자원 관리 시스템(remote resource management system)에 있어서,

대칭적 멀티프로세서 시스템의 클러스터 노드들(cluster nodes)간에 인터페이스를 가진 대칭적 멀티프로세서의 다수의 클러스터와,

로컬 인터페이스(local interface) 및 인터페이스 제어기(interface controller)와,

제각기 로컬 인터페이스 제어기를 가진 하나 이상의 원격 저장 제어기(one or more remote storage controller)와,

로컬-원격 데이터 버스(local-remote data bus)와,

대칭적 멀티프로세서들의 두 클러스터간의 인터페이스를 관리하기 위한 원격 자원 관리기(remote resource manager) - 상기 두 클러스터의 각각은 다수의 프로세서, 공유 캐쉬 메모리(shared cache memory), 다수의 I/O 아답터(adapter) 및 클러스터로부터 액세스 가능한 메인 메모리(main memory accessible from the cluster)를 가짐 -

를 구비하며, 상기 원격 자원 관리기는 원격 저장 제어기를 가진 자원들을 관리하여 그 원격 제어기에 작업을 분배하며, 이 제어기는 작업 요구를 개시한 요구기(requester)를 알 필요 없이 원하는 동작을 수행하는 대리 프로세스(agent)로서 작용하고, 상기 작업은 대칭적 멀티프로세서들의 상기 클러스터들간의 일정한 통신을 필요로 하지 않고서도 원격 요구기가 그 작업을 처리하는데 이용가능할 때에만 전달되는 원격 자원 관리 시스템.

청구항 2. 제1항에 있어서, 상기 원격 자원 관리 시스템은 각 클러스터상의 단일 인터페이스 매크로(single interface macro)를 가지며, 이 단일 인터페이스 매크로는 대기 요구들(queued requests)을 우선순위화하고(prioriting) 새로운 동작들을 인터페이스를 통해 전송하고 다른 사이드로부터의 복귀 응답을 처리하며 클러스터들간에서의 모든 데이터 전송을 감독하는 것을 포함하는 인터페이스 임무들의 제어를 담당하며, 상기 로컬 인터페이스 제어기는 원격 사이드에 대해 작업 요구를 개시할 뿐만 아니라 원격 사이드상의 인출/저장 제어기(fetch/store controller)를 관리하여, 이용가능한 원격 제어기에 새로운 동작을 즉시 라우팅(routing)해서, 상기 원격 인출/저장 제어기가 단순히 상기 로컬 인터페이스 제어기 대신에 작업을 행하는 대리 프로세스로서 되게 하고, 상기 로컬 인터페이스 제어기가 요구기 대신에 작업을 행하게 함으로써, 동작의 오너(owner of the operation)를 식별하는 정보를 보낼 필요가 없게 하는 원격 자원 관리 시스템.

청구항 3. 제2항에 있어서, 수개의 로컬 동작이 단일의 단위(atomic) 원격 동작으로 조합될 수 있게 하는 명령 재맵핑 동작(command remapping operation)을 갖는 원격 자원 관리 시스템.

청구항 4. 제3항에 있어서, 판독 전용 데이터 카피(read-only copy of data)를 위한 프로세서 인출 요구(processor fetch request) 및 저장 보호 키(storage protection key)를 포함하는 판독 전용 데이터(read-only data)를 위한 인출 요구는 동일한 상태도(identical state diagrams) 및 캐쉬 관리 동작을 이용하기 위해 상기 원격 클러스터상에 인출 제어기를 필요로 하며, 상기 인터페이스 제어기는 그들 양자를 판독 전용 라인 인출(Read Only Line Fetch)로서 알려진 간단한 원격 저장 클러스터(RSC IC) 인터페이스 제어기 명령으로 재맵핑시켜, 상기 원격 저장 클러스터(RSC) 인터페이스 제어기에 의해 처리되어야 하는 동작들의 수를 감소시키는 원격 자원 관리 시스템.

청구항 5. 제4항에 있어서, 저장 데이터의 전송이 로컬-원격 데이터 버스를 불필요하게 구속하며, 부가적인 제어 라인들이 디렉토리 정보(directory information)를 전송하기 위해 필요한 경우, 상기 인터페이스 제어기는 디렉토리 상태에 따라 전달 명령들을 "강제 방출(force cast out)" 및 "판독 전용 무효화(read-

only invalidate)" 명령으로 재맵핑하는 원격 자원 관리 시스템.

청구항 6. 제4항에 있어서, 상기 원격 자원 관리 시스템은 하나 이상의 파이프라인된 계층적 레벨 캐쉬(pipelined hierarchical cache)에 대해 서비스를 제공하는 다수의 인출 및 저장 제어기를 포함하는 하이-엔드(high-end) 저장 서브시스템과의 인터페이스를 구비하며, 일련의 우선순위 스테이션(priority station)은 상기 인터페이스를 통해 전송하기 위한 요구를 선택하며, 다수의 파이프가 관련되는 경우에 각 파이프내의 사전 우선순위 스테이션(pre-priority station)은 RSC IC로 전송하기 위한 인출 또는 저장 요구를 선택하며, 동일 사이클 동안 상기 원격 저장 제어기의 인터페이스 제어기는 명령 유형 (command type) 및 자원 가용성(resource availability)에 근거하여 최적의 요구를 선택하기 위해 우선순위 동작을 이용하는 원격 자원 관리 시스템.

청구항 7. 제6항에 있어서, 상기 우선순위 동작에 대해, 다수의 파이프가 소정의 사이클에서 인터페이스 사용을 요구할 수 있으므로 원격 인출 제어기가 이용가능한 동안에는 동작은 저장보다는 인출을 지지하며, 그렇지 않으면 원격 저장 제어기가 이용가능하고 또한 데이터 경로가 그를 필요로 하는 저장 동작에 대해 이용가능한 동안에는 저장이 취해지며, 그들 두 요구가 인출이고 그들 두 요구가 이용가능한 자원을 갖는 경우에는 어떤 요구를 우대할 것인지의 여부를 간단한 라운드 로빈(round robin)에 의해 판단하나, 그들 두 요구가 저장인 경우에는 어느 파이프가 이용가능한 자원을 갖는 가에 의해 승자(winner)가 결정되며, 그들 둘 모두가 이용가능한 자원을 갖는 경우에는 간단한 라운드 로빈이 사용되는 원격 자원 관리 시스템.

청구항 8. 제6항에 있어서, 원격 사이드상에 대기되고 있는 작업을 전송하는 인터페이스 사이클이 허비되지 않게 하는, 각 로컬 인터페이스 제어기내의 원격 자원의 관리기를 구비하는 원격 자원 관리 시스템.

청구항 9. 제6항에 있어서, 각 원격 저장 제어기의 로컬 인터페이스 제어기는 동기식 및 비동기식 응답 버스를 이용하여 캐쉬 코히어런스(cache coherency)를 유지하는 한편 성능을 극대화하며, 비동기식 응답 버스는 원격 동작의 공식적 종료를 표시하고 오리지널 요구기(original requester)로 전송되기도 하는 모든 최종 응답에 대해 사용되며, 상기 최종 응답에는 로컬 디렉토리가 정확한 최종 상태로 갱신될 수 있게 하는 변경 라인 정보(change line information)가 붙여지는 원격 자원 관리 시스템.

청구항 10. 제9항에 있어서, 상기 원격 제어기의 인터페이스 제어기는 모든 클러스터-클러스터 데이터 흐름을 관리하고 상기 로컬 저장 제어기로부터의 요구들을 상주하는 원격 인출 제어기로부터의 요구들과 비교하여 인출 데이터를 복귀시키며, 그들 요구가 데이터 경로를 차지하기 위해 서로 경쟁하는 사이클 동안 우선권(preference)이 복귀 인출 데이터에 주어지며, 인출 데이터가 원격 메인 메모리로부터 획득되는 경우 상기 원격 제어기의 인터페이스 제어기 관리기는 대응하는 데이터 경로를 메모리 뱅크로부터의 데이터 액세스 시에 감시 및 관리하며, 상기 원격 저장 제어기 데이터 경로가 이용가능한 경우 데이터는 원격 인출 버퍼를 우회함으로써, 데이터의 임시적인 버퍼링과 연관된 통상의 대기 시간이 감소되는 원격 자원 관리 시스템.

청구항 11. 제9항에 있어서, 전체적인 시스템 처리능력을 향상시키도록 복제된 (replicated) 원격 저장 제어기 자원들의 관리를 향상시키기 위해, 상기 원격 제어기의 인터페이스 제어기 관리기는 복제 원격 인출 자원들 간에 작업 요구들을 교번적으로 사용하는 것에 의해 상기 원격 캐쉬에 대한 적중을 발생하는 연속적인 인출 요구들을 관리하고, 이용가능한 경우 복제 원격 인출 제어기에 제 2의 인출 요구를 전송함으로써, 상기 복제 원격 인출 제어기는 그의 버퍼에 대한 로딩을 개시할 수 있는 한편 제 1 원격 제어기 버퍼는 그의 데이터 전송을 완료하므로, 상기 제 1 버퍼의 전송 완료시에 상기 제 2 버퍼는 그의 데이터를 즉시 인터페이스를 통해 전송할 수 있는 원격 자원 관리 시스템.

청구항 12. 제9항에 있어서, 상기 원격 제어기의 인터페이스 제어기 관리기는 크로스-클러스터 데드록(cross-cluster deadlock)을 유발할 수 있는 동작 시퀀스를 감시하도록 구성된 데드록 회피 메카니즘을 구비하며, 이러한 시나리오의 검출시 상기 원격 제어기의 인터페이스 제어기 관리기는 특정 거절 응답(special reject response)을 개시 클러스터(initiating cluster)에 다시 복귀시킴으로써 현안중의 동작(pending operation)을 거절할 것이며, 상기 원격 제어기의 인터페이스 제어기는 그 거절을 발신(originating) 인출/저장 제어기로 전송하여 동작이 재시도될 수 있게 하고 데드록 윈도우(deadlock window)가 사라질 때까지 연속적으로 거절되고 재시도될 수 있게 하는 원격 자원 관리 시스템.

청구항 13. 제9항에 있어서, 인터페이스 패리티 에러(interface parity error)가 새로운 원격 저장 제어기 동작을 수반하는 어떤 제어 정보에서 검출되는 경우 동기식 인터페이스는 명령 전송후 일정 수의 사이클내에서 인터페이스 에러 상태를 전송하는데 사용되며, 에러의 경우 상기 발신 인출/저장 제어기는 그 사실을 통보받고 복구의 적격성을 판단하며, 상기 원격 저장 제어기의 인터페이스 제어기는 대응하는 원격 저장 제어기의 자원을 자동적으로 리세트시켜 동작이 다시 요구될 수 있게 하는 원격 자원 관리 시스템.

청구항 14. 제1항에 있어서, 상기 시스템은 명령들의 세트를 더욱 효과적인 작은 서브세트로 재맵핑함으로써 상기 원격 제어기의 복잡성을 감소시키고 불필요한 데이터 전송 방지에 의해 인터페이스 효율을 향상시키는 명령 재맵핑 수단을 포함하는 원격 자원 관리 시스템.

청구항 15. 제14항에 있어서, 상기 명령 재맵핑은 플립 비트(flip bits)를 사용하여 필요한 수의 게이트를 감소시키고 또한 기능이 단일 로직 사이클내에서 수행될 수 있게 하여 시스템 성능을 향상시키는 원격 자원 관리 시스템.

청구항 16. 제6항에 있어서, 상기 시스템은 로컬 개시 동작(locally initiated operations)의 처리를 위한 통합된 제어(controls) 및 원격 복귀(remote returns)를 구비하여, 아웃바운드 및 인바운드(outbound and inbound) 데이터가 상기 데이터 버스를 공유할 수 있게 하는 것에 의해, 인터페이스 I/O를 감소시키고 또한 상기 버스가 전체적인 시스템 성능에 관련하여 고효율적인 방식으로 관리될 수 있게 하는 원격 자원 관리 시스템.

청구항 17. 제6항에 있어서, 상기 우선순위 동작은, 원격 개시 동작(remote initiated operations)에 대한 로컬 개시 요구 및 응답과 데이터 경로를 필요로 하는 동작에 대한 데이터 경로 가용성(availability)을 고려하여 저장 요구보다 인출 요구를 지지함으로써 또한 동작들이 적정한 자원이 이용가능한 경우에 인터페이스를 통해 전송될 수 있게 함으로써, 인터페이스 이용에 의해 시스템 성능을 효율적으로 향상시킬 목적으로 매 사이클에서 요구들 및 원격 저장 제어기 자원들을 동적으로 분석하는 원격 자원 관리 시스템.

청구항 18. 제1항에 있어서, 상기 원격 자원 관리 관리기는 원격 캐쉬에서의 디렉토리 적중실패(directory miss)가 발생하는 경우에 일정 시간내에 자원이 자동적으로 해제될 수 있게 하는 동기식 교차 질의(synchronous cross interrogate)와 CP 인출에 대한 파이프 신속-경로화(pipe fast-pathing)를 제공하며, 상기 파이프 신속-경로화는 RSC IC가 파이프라인을 감시하여 CP 인출을 찾아보고, 발견한 때에 그 인출을 RSC IC에 제시하기 전에 LFAR 제어기내로 로딩해야 하는 정상적인 경우보다 1 사이클 앞서 그 인출의 개시를 시도하며, 조기 PMA 인출 w/제거(early PMA w/Cancel) - 이 조기 PMA 인출 w/제거에서, 로컬 조기 PMA는 교차 질의가 인터페이스를 통해 전송됨과 동시에 데이터를 인출하기 시작하나 원격 캐시에서 적중이 발생되는 경우에는 상기 RSC IC가 로컬 LFAR 제어기에게 상기 조기 PMA 인출을 제거하여 메모리 인터리브(memory interleaves)를 자유롭게 하도록 함 - 와 계층적 캐쉬 액세스(hierarchical cache access)시의 인출 버퍼 우회(Fetch Buffer bypass)를 지원하도록 하며, 상기 RSC IC는 데이터가 계층적 캐쉬(PMA)로부터 수신되고 있는 동안 클러스터-클러스터 데이터 경로를 감시하여 그 데이터 경로가 이용가능하면 그 데이터가 자동적으로 인출 버퍼를 우회하여 PMA 수신 포트로부터 직접 RSC 인터페이스로 흐르게 하는 원격 자원 관리 시스템.

청구항 19. 제1항에 있어서, 상기 원격 자원 관리 관리기는 단일의 크로스포인트 제어기를 사용하여 4-웨이 동시 데이터 전달이 가능한 4개의 데이터 경로 - 이들 데이터 경로의 각각은 로컬 개시 및 원격 개시 동작들을 다중화함 - 를 관리하여, 그들 데이터 경로의 가용성이 디스패칭을 위한 다음 동작의 결정시 우선순위 메카니즘에 전송되도록 하는 원격 자원 관리 시스템.

청구항 20. 제1항에 있어서, 상기 원격 자원 관리기는 상기 원격 사이드가 판독 전용 무효화에 필요한 모든 단계를 완료하기 전에 상기 로컬 LFAR 제어기가 해제될 수 있게 함으로써 이후의 판독 전용 무효화에서도 상기 LFAR 제어기가 원격 동작 전송을 포함하는 새로운 동작을 자유롭게 시작할 수 있도록 하는 가속 판독 전용 무효화 동작(an accelerated Read Only Invalidate)과,

관련된 LFAR 또는 LSAR 제어기의 통지(the notification of the associated LFAR or LSAR controller)를 비롯한 인터페이스 패리티 에러의 경우에 RSC 자원이 자동적으로 리세트될 수 있게 함으로써 원하는 경우에 상기 제어기가 동작을 재시도할 수 있게 하는 동기식 인터페이스 검사의 사용(use of a synchronous interface check)과,

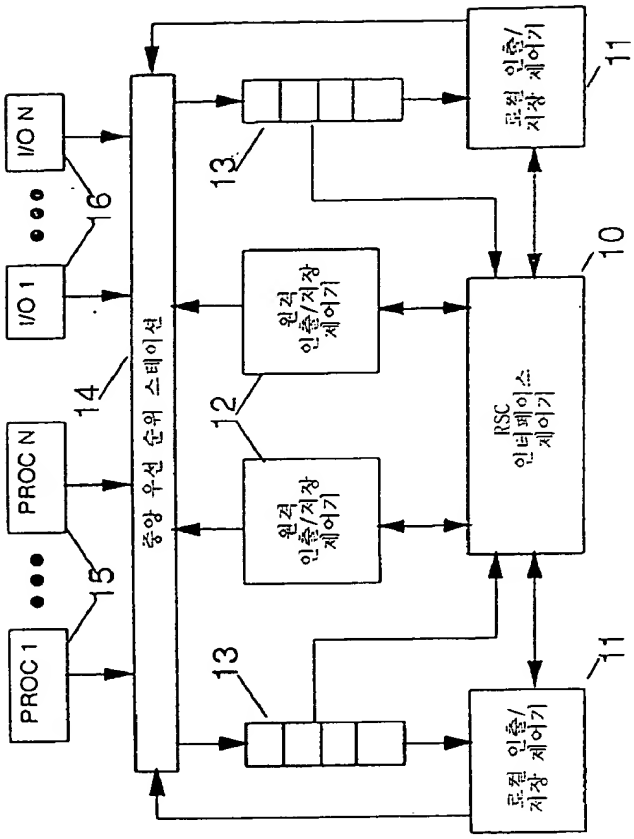
상기 원격 RFAR 또는 RSAR 제어기가 잠재적인 데드록을 검출하여 거절 응답 - 이 거절 응답은 대응하는 LFAR 또는 LSAR 제어기에 전송됨 - 을 전송해서 상기 제어기가 동작을 재시도할 수 있게 하는 크로스 클러스터 데드록 회피 수단(means for cross cluster deadlock avoidance)과,

한 쌍의 RFAR 또는 RSAR 자원을 구성하는 양 구성원들(both members)이 이용가능한 경우에 연속적인 데이터 인출(the consecutive data fetches)이 그 쌍의 양 구성원에 교대로 분배되게 함으로써, 상기 인출중의 후자의 인출(the latter fetch)은 하나의 원격 버퍼에 로딩되기 시작하는 한편 전자의 인출(the former fetch)의 트레일링 바이트(trailing bytes)는 다른 버퍼에서 여전히 처리되도록 하는 쌍 RSC 자원의 사용(use of Paired RSC Resource)

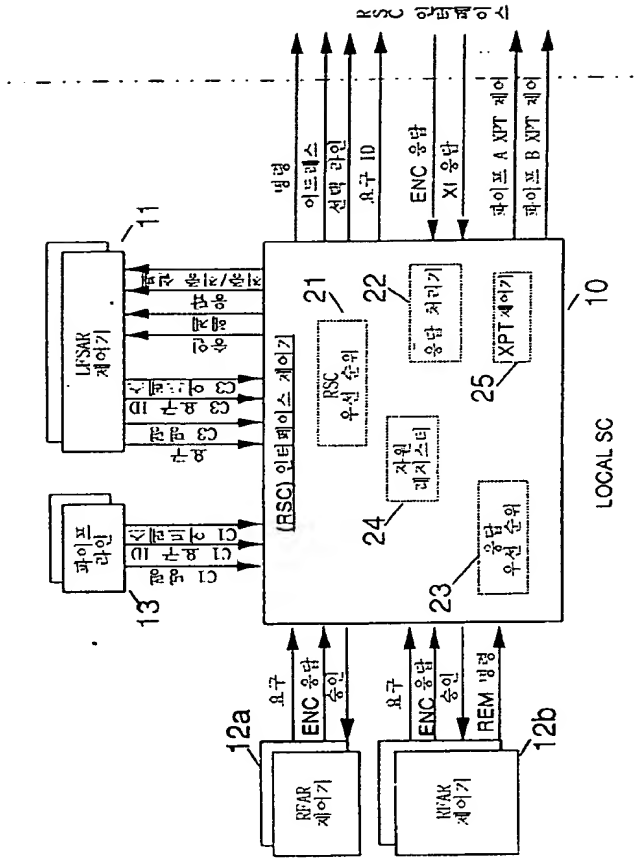
을 제공하는 원격 자원 관리 시스템.

도면

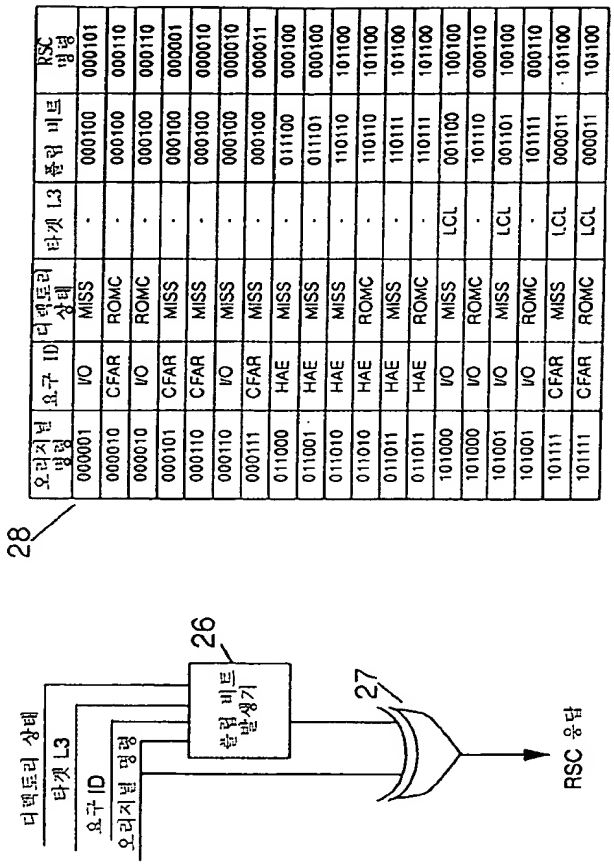
도면1a


I/O N
I/O 1
PROC N
PROC 1
16
15
14
중앙 우선 순위 스테이션
13
12
원격 입출/저장 제어기
RSC 인터페이스 제어기
10
11

도면1b

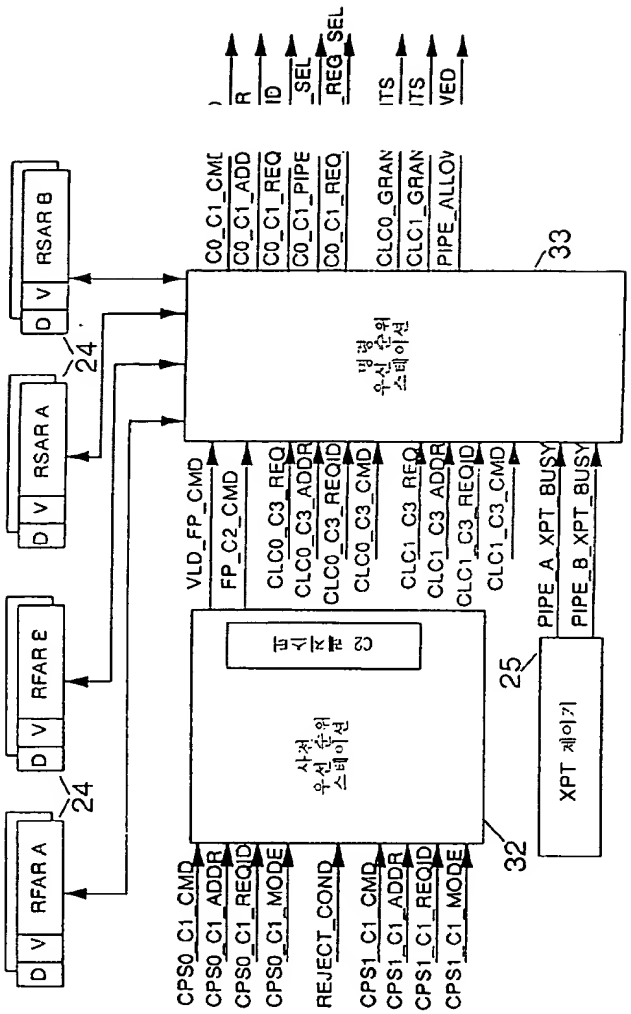

RSC 상태정보
명령
어드레스
선택 라인
요구 ID
ENC 응답
XI 응답
패이프 A XPT 제어
패이프 B XPT 제어
11
LFSAK 제어기
응답/지연 선택
응답
선택
C3 응답 ID
C3 명령
요구
패이프 라인
C1 선택
C1 요구 ID
C1 명령
13
(RSC) 인터페이스 제어기
21
RSC 우선 순위
22
응답 처리기
25
XPT 제어기
10
LOCAL SC
24
자원 레지스터
23
응답 우선 순위
요구
ENC 응답
승인
12a
RFAR 제어기
요구
ENC 응답
승인
REM 명령
12b
RFAK 제어기

도면2a

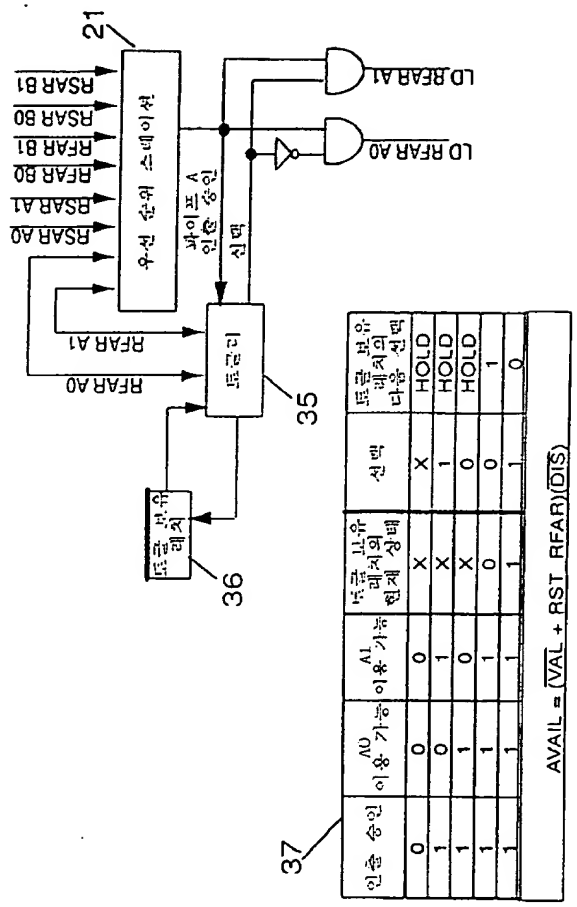


오리지널 명령	요구 ID	디렉토리 상태	타겟 L3	폴링 비트	RSC 명령
000001	I/O	MISS	-	000100	000101
000010	CFAR	ROMC	-	000100	000110
000010	I/O	ROMC	-	000100	000110
000101	CFAR	MISS	-	000100	000001
000110	CFAR	MISS	-	000100	000010
000110	I/O	MISS	-	000100	000010
000111	CFAR	MISS	-	000100	000011
011000	HAE	MISS	-	011100	000100
011001	HAE	MISS	-	011101	000100
011010	HAE	MISS	-	110110	101100
011010	HAE	ROMC	-	110110	101100
011011	HAE	MISS	-	110111	101100
011011	HAE	ROMC	-	110111	101100
101000	I/O	MISS	LCL	001100	100100
101000	I/O	ROMC	-	101110	000110
101001	I/O	MISS	LCL	001101	100100
101001	I/O	ROMC	-	101111	000110
101111	CFAR	MISS	LCL	000011	101100
101111	CFAR	ROMC	LCL	000011	101100

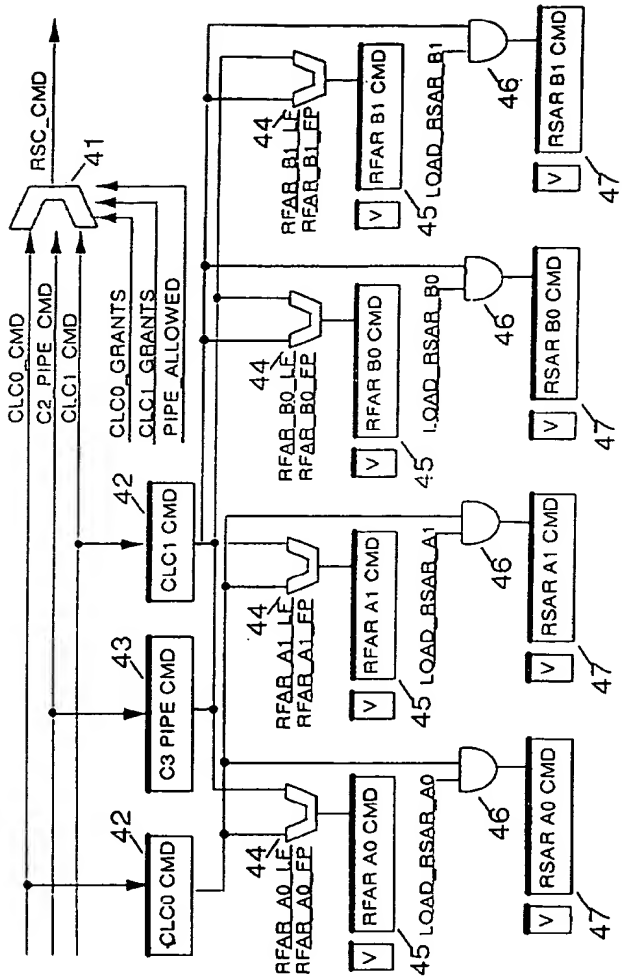
도면3a


RSAR B
RSAR A
RFAR B
RFAR A
D V
24
33
32
25
C0_C1_CMD
C0_C1_ADDR
C0_C1_REQ
C0_C1_PIPE
C0_C1_REQ
CLC0_GRAN
CLC1_GRAN
PIPE_ALLOW
R
ID
_SEL
_REG
SEL
ITS
ITS
VED
VLD_FP_CMD
FP_C2_CMD
CLC0_C3_REQ
CLC0_C3_ADDR
CLC0_C3_REQID
CLC0_C3_CMD
CLC1_C3_REQ
CLC1_C3_ADDR
CLC1_C3_REQID
CLC1_C3_CMD
PIPE_A_XPT_BUSY
PIPE_B_XPT_BUSY
CPS0_C1_CMD
CPS0_C1_ADDR
CPS0_C1_REQID
CPS0_C1_MODE
REJECT_COND
CPS1_C1_CMD
CPS1_C1_ADDR
CPS1_C1_REQID
CPS1_C1_MODE
XPT 제어기

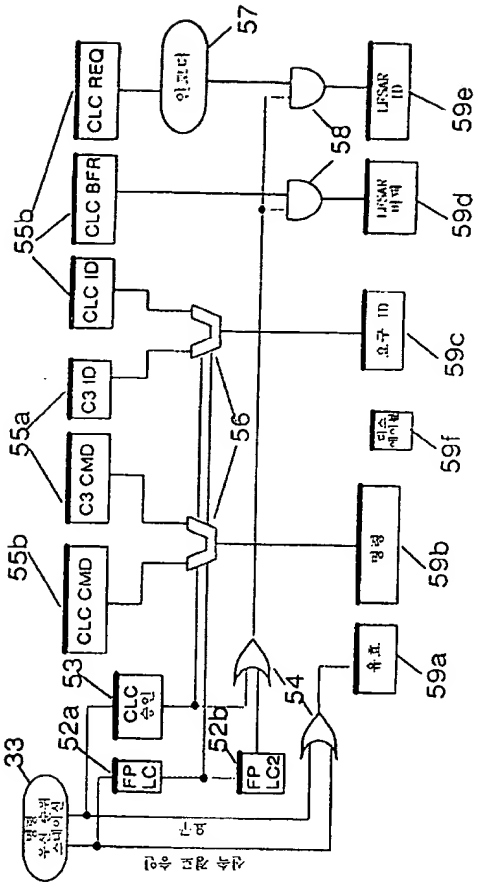
도면3b


21
우선 순위 스테이션
RSAR B1
RSAR B0
RFAR B1
RFAR B0
RSAR A1
RSAR A0
RFAR A1
RFAR A0
LD RFAR A0
LD RFAR A1
A 인출 승인
신택
프로그램
35
프로그램 보유 레지스터
36
37
인출 승인
A0 이용 가능
A1 이용 가능
현재 보유 레지스터 상태
신택
프로그램 보유 레지스터 다음 신택
HOLD
HOLD
HOLD
AVAIL = VAL + RST RFAR)(DIS)

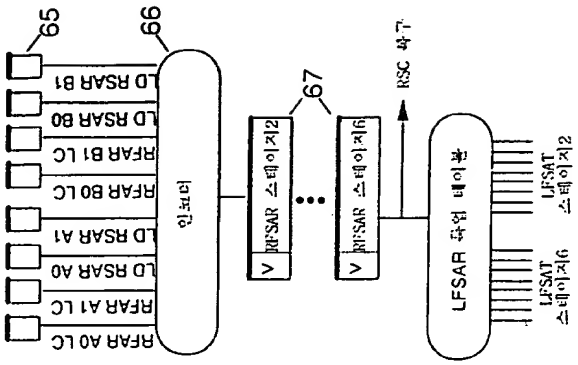
도면4

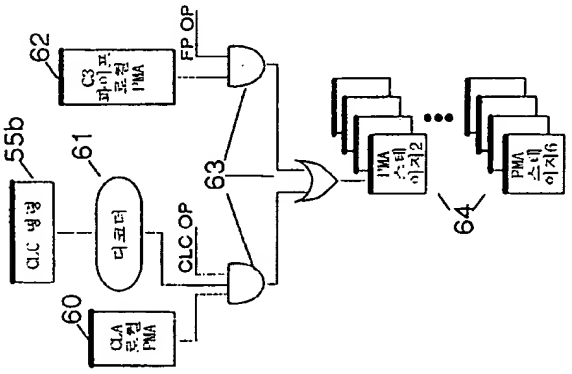

RSC_CMD
41
CLC0_CMD
C2_PIPE_CMD
CLC1_CMD
CLC0_GRANTS
CLC1_GRANTS
PIPE_ALLOWED
42
CLC1 CMD
43
C3 PIPE CMD
42
CLC0 CMD
44
RFAR_A0_LE
RFAR_A0_EP
RFAR A0 CMD
V
45
LOAD_RSAR_A0
46
RSAR A0 CMD
V
47
44
RFAR_A1_LE
RFAR_A1_EP
RFAR A1 CMD
V
45
LOAD_RSAR_A1
46
RSAR A1 CMD
V
47
44
RFAR_B0_LE
RFAR_B0_EP
RFAR B0 CMD
V
45
LOAD_RSAR_B0
46
RSAR B0 CMD
V
47
44
RFAR_B1_LE
RFAR_B1_EP
RFAR B1 CMD
V
45
LOAD_RSAR_B1
46
RSAR B1 CMD
V
47

도면5

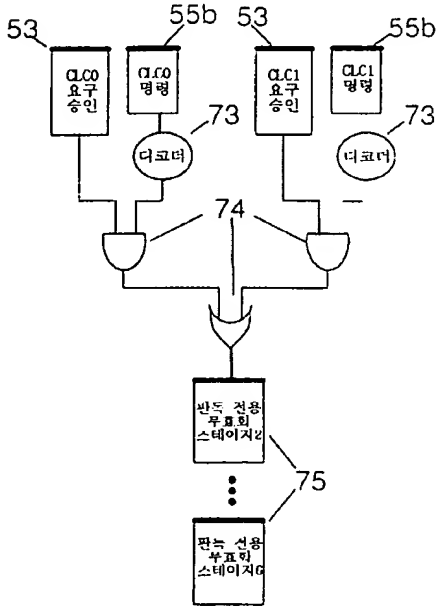

57
인코더
CLC REQ
CLC BFR
CLC ID
C3 ID
C3 CMD
CLC CMD
55b
55a
55b
58
59e
59d
59c
59f
59b
59a
56
53
54
52a
52b
33
CLC 승인
FP LC
FP LC2
요구
상태
명령
요구 ID
링크 에러
LRSNR ID
LRSNR 비트
상태 정보 송신
명령 수신 스테이션

도면6

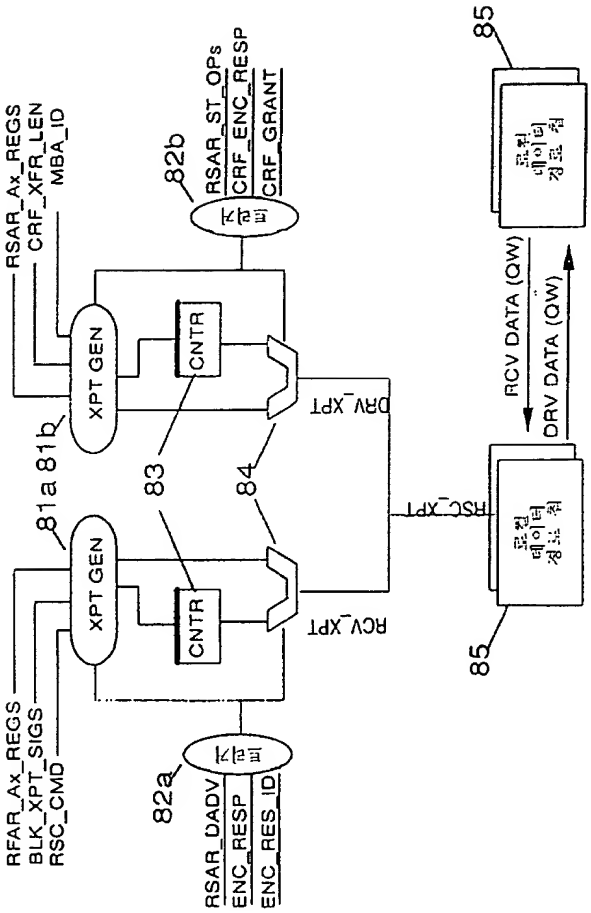

65
66
67
LD RSAR B1
LD RSAR B0
RFAR B1 LC
RFAR B0 LC
LD RSAR A1
LD RSAR A0
RFAR A1 LC
RFAR A0 LC
인코더
RFSAR 스테이지2
RFSAR 스테이지6
RSC 출구
LFSAR 유입 데이블
LFSAT 스테이지2
LFSAT 스테이지6


62
C3 파이프 포함 PMA
FP OP
55b
61
63
CLC 명령
디코더
CLC OP
PMA 스테이지2
PMA 스테이지6
64
60
CLA 명령 PMA

노면7


53
55b
53
55b
CLC0 요구 승인
CLC0 명령
CLC1 요구 승인
CLC1 명령
73
73
디코더
디코더
74
판독 전용 무표회 스테이지2
판독 전용 무표회 스테이지6
75

도면8


RSAR_Ax_REGS
CRF_XFR_LEN
MBA_ID
RSAR_ST_OPs
CRF_ENC_RESP
CRF_GRANT
82b
트리거
85
로컬 메이터 경로 제
XPT GEN
CNTR
81b
DRV_XPT
RCV DATA (QW)
DRV DATA (QW)
81a
83
84
RSC_XPT
RCV_XPT
로컬 메이터 경로 제
85
RFAR_Ax_REGS
BLK_XPT_SIGS
RSC_CMD
82a
트리거
RSAR_DADV
ENC_RESP
ENC_RES_ID